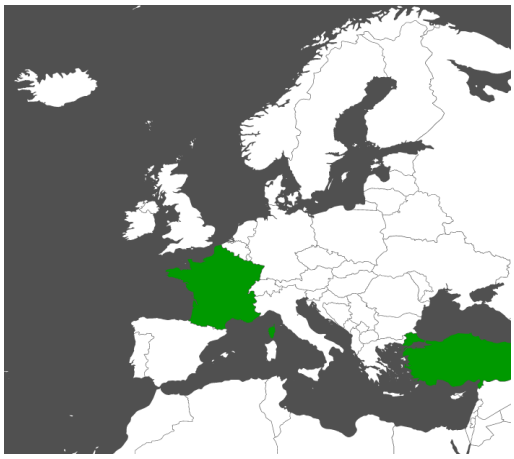
ADAX project
Attack Detection and
Counter measure Simulation

WP2 D2.1 v1.1
-
State-of-the-art intrusion detection, prevention and reaction simulation systems analysis report

2014-08-01



History

Version	Date	Description, Author(s)
1.0	2013-04-15	BOUN,CCS, E1A1, IT, SCR, NET, PRO, YAP
1.1	2014-08-01	Public version, BOUN,CCS, E1A1, IT, SCR, NET, PRO, YAP

Participants in the project

Participant name	Short name
Cassidian CyberSecurity SAS	CCS
Boğaziçi University	BOUN
Plus One Minus One	E1A1
Institut Mines Telecom	IT
6cure	SCR
Netasq	NET
Provus	PRO
Yapı Kredi Bank	YAP

Table of content

Table of Figures

1. Executive summary

In today's market structure there is a wide gap between the detection/supervision and reaction products. The goal of ADAX project is to fill this gap, by first collecting needs of ICT system operators and then establishing a state-of-the-art of techniques and existing tools. The main outcome will be the identification and realization of a set of expected requirements to enable the impact assessment through the simulation of countermeasure enforcement.

This report is an ADAX project deliverable defined as *D2.1, intrusion detection, prevention and reaction simulation systems: state-of-the-art analysis report* and prepared to describe the state-of-the-art for the technical and scientific topics related to ADAX.

This document is organized as follows. In Chapter 2, we provide a glossary containing the terminology. The introduction to the report topic is given in Chapter 3. Attack types are described in Chapter 4. The state-of-the-art for intrusion detection systems (IDSes) are presented in Chapter 5. The concept of intrusion prevention and reaction is given in Chapter 6. The simulation environment for the target IDS system is explained in Chapter 7. Finally, conclusions are presented in Chapter 8. Product description, requirements and an overview of the existing products are provided in a separate deliverable report.

2. Glossary

ADAX	Attack detection and counter measure simulation
CCC	Common control channel
CSI	Compound signature identification
DDoS	Distributed denial of service
IDS	Intrusion detection system
IPS	Intrusion prevention systems
ROC	Receiver operating characteristic
SIEM	Security information and event management
SOS	Secure overlay service
SSDF	Spectrum sensing data falsification

3. Introduction

This document aims at looking into current technical/scientific state-of-the-art in the relevant domains for the ADAX project, namely intrusion detection/prevention and alert correlation, countermeasures and reaction and network activity simulation.

ADAX (Attacks Detection and Countermeasures Simulation) is a 30-month project that aims to study feasibility of solutions enabling to detect complex attacks against an information system working in its complex environment and to react smartly and quickly to those attacks with adopted countermeasures.

The core innovation of the ADAX project transferred to industry is a decision-support system for security operations and policy management integrated within a security information management platform, interacting with alert correlation systems, acting as a mediator between the subscriber identity module (SIM) environment and the monitored information communication technologies (ICT) system, helping the operator to assess the seriousness of security issues, validate the remediation actions and reactions, deploy them over the monitored infrastructure, and monitor their efficiencies. This prototype will include the ability to mitigate threats at the network and at the service layer.

In this first deliverable of the project, we provide an in depth literature review and present the state-of-the-art for the technical and scientific topics related to ADAX..

3.1 Motivation

ICT system operators need to rapidly remediate to intrusions or vulnerabilities detected in an information system by selecting countermeasures. Unfortunately, it is currently impossible to assess the impact of a reaction since there is no available quantitative assessment of the attacks.

ADAX will define solutions enabling proposal of reactions and means to assess impact of countermeasures before their enforcement on the information system security and on the services running above.

3.2 Target Markets

ADAX is addressing three markets as listed below:

- Security information and event management (SIEM) (on the detection/supervision side),
- Intrusion prevention systems (IPS)
- Distributed denial of service (DDoS) (both on the reaction side)

In current market structure, there is a wide gap between the detection/supervision and reaction products. Furthermore, the existing reaction products focus on very limited set of countermeasures, and furthermore these countermeasures are deployed without any decision support. Details about the target market analysis and products are available in ADAX-Full Project Proposal Document (ITEA 2 Full Project Proposal Annex, ip10030, Oct. 2010).

3.3 Project Goals

ADAX aims at addressing the following issues in order to help operators to assess the seriousness of a set of currently active threats and the impact of the reactions on the monitored ICT system:

- Large-scale modeling of information systems and networks,
- Efficient alert correlation,
- Quantitative evaluation of simulation results for decision support,
- Study of the combination of multiple attacks and countermeasures,
- Development of novel visual analytics technologies for the identification and prediction of very complex patterns of abnormal situation in the network,
- Effective deployment of the selected countermeasures, and trust and security in future Internets.

Two main use cases are envisaged for the project:

- The detection of an attack is followed by the simulation of a countermeasure proposed by a decision-making tool.
- The simulation environment must allow an operator to check that the countermeasure does not affect the system or at least is not worse than the attack itself.
- After an audit, it has been proved that an element of the system under monitoring is vulnerable to a kind of attack.

The solution proposed by the project must permit, through a simulation, to know the impact of a countermeasure on the security of running services.

3.4 Similar projects

ITEA 2 stimulates and supports innovative, industry-driven, pre-competitive research and development (R&D) projects that will contribute research excellence to Europe's competitive software-intensive systems and services sector.

The ITEA 2 approach is;

- Industry driven;

- Bottom-up to favor innovation;
- Flexible to favor business impact;
- Market oriented;
- Intergovernmental;
- Based on a multi-dimensional concept of excellence for project selection; and
- Community oriented, easily accessible by industry and SMEs.

Security threats are inherently difficult to manage because they are constantly evolving. Security requirements are thus shifting to address the full spectrum of risks and threats, integrating new dimensions such as intelligence and surveillance and evolving towards more resilient, dynamic and scalable concepts.

In 2010, the global information security market hit a value of US$70 billion, and by 2016 this figure is expected to exceed a market value of US$85 billion. Thus, security market oriented projects have more places in ITEA 2, with the aim of being industry driven, market oriented and flexible to favour business impact.

The projects serve the management, operations, and planning levels of an organization and help to make decisions, which may be rapidly changing and not easily specified in advance, are the other market-oriented projects. The projects, consists of information technology, marketing data and modeling capabilities that enable the system to provide predicted outcomes from different scenarios and marketing strategies, so answering "what if?" questions, have found places in ITEA 2.

Due to above mentioned ADAX aim is fill the gap between the detection/supervision and reaction products and support mechanisms helping the security operators to make enlightened decisions in a dynamic situation.

The running and completed security and decision support related projects are listed below.

3.4.1 Running projects

Diamonds (Development and Industrial Application of Multi-Domain Security Testing Technologies)

The ITEA 2 DIAMONDS project is developing a new, model-based approach to software testing that could form the basis of a new standard in formal security testing of software systems. Efficient and automated security testing methods of industrial relevance for highly secure systems in multiple domains (incl. e.g. banking, transport or telecommunication).

ATAC (Advanced Test Automation for Complex Software-Intensive Systems)
The ATAC-project aims at the development of automated test approaches for effective and efficient quality assurance of complex and highly configurable systems. This will be done by re-using existing mature techniques and providing a systematic and tool-supported quality assurance process.
Analyze the current situation for system quality assurance

- Develop and enhance high performance methods and tools
- Deploy methods and tools for quality assurance of large and complex software-intensive systems

3.4.2 Completed projects

The TECOM (Trusted Embedded Computing)

The TECOM project has developed architectures and solutions combining embedded trust services and trusted operating system technologies to ensure security and dependability in a wide range of complex and dynamic embedded systems. The project focused on enabling multiple applications to be run safely on the same systems and processors while acting totally independently of each other.

The TECOM (Trusted Embedded Computing) project has developed architectures and solutions combining embedded trust services and trusted operating system technologies to ensure security and dependability in a wide range of complex and dynamic embedded systems. The project focused on enabling multiple applications to be run safely on the same systems and processors while acting totally independently of each other.

Applications range from protecting film rights in video-on-demand applications to ensuring bug-free software upgrades in domestic appliances.

SATURN (Security Applications and Technologies for Universal infoRmation Networks)

In recent years, Internet use has continued to grow in the business environment and now roughly 90 percent of European businesses use it for e-mail, browsing the web and hosting a web site. Furthermore, most web sites now enable customers to initiate electronic transactions. There has been a massive increase in the use of WLAN network and nomadic solution. This greater connectivity has increased the business' exposure to security threats, which continue to evolve.

PubSub4RT (A publish/Subscribe Infrastructure for Real-Time Services)

PubSub4RT aims at development of a middleware infrastructure for processing massive data flows making use of publish/subscribe technology. Many applications in communications, banking, security, Internet services and sensor networks involve treating massive amounts of data over wide geographic areas. The ITEA 2 PubSub4RT project has developed an infrastructure to handle such flows in real time using publish/subscribe technology. This opens up a range of novel services, particularly in the financial sector, business intelligence and telecommunications. The platform is highly scalable and has been successfully demonstrated in wide-area networks for real-time credit-card fraud detection.

EUROSYSLIB (European Leadership in System Modeling and Simulation through advanced Modelica Libraries)

The ultimate objective of EUROSYSLIB is to make Modelica the de-facto standard language for embedded system modeling and simulation. In order to support this major product lifecycle management effort, the EUROSYSLIB consortium, composed of 18 European partners, is committed to delivering a large set of high-value, innovative modeling and simulation libraries based on the freely available Modelica object-oriented modeling language.

Managing increasing complexity in embedded software while cutting time to market and improving product quality is essential. The ITEA 2 EUROSYSLIB project has dramatically reinforced European leadership in systems modeling and simulation through enhancement of the Modelica modeling language, its accompanying libraries and infrastructure. EUROSYSLIB developed support for multi-domain applications in aircraft systems, power plants, conventional and electric vehicles with interoperability between toolboxes and a huge extension of the Modelica libraries. The results are available through the Modelica Association and commercial tool suppliers.

MODELISAR

The purpose of MODELISAR is to introduce functional mock-up (FMU), a next generation of methods, standards and tools to support collaborative design, simulation and test of systems and embedded software.

EDAFMIS (Embedded Decision and Data Fusion for Medical Intervention Support)

Information technology has a key role in the operating theatre to support the trend to minimally invasive surgery.
The ITEA 2 EDAFMIS project has enabled equipment interoperability, improved real-time imaging technology, and simplified communications with external colleagues and speeded access to expert information. As a result, surgeons can work faster, allowing patients to go home earlier and get back to work sooner, as well as helping avoid medical errors.

3D Test Bench

To enable all stakeholders in the design process of complex systems to naturally and intuitively be involved to the best of their abilities, collaboratively with other stakeholders, enabling innovative concepts, improved quality products, reduced time to market and reduced design and development costs.

Establish a product development process and workflow & the roles of all stakeholders, in the modeling and simulation supported product development lifecycle

- Implement Domain Specific Language (DSL) and supporting technology for specifying, analyzing, designing, and verifying complex multi-disciplinary systems,
- Establish a simulation and modeling environment that supports collaborative test driven virtual product development.

MULTIPOL

MULTIPOL provides users of independently administered security domains with the ability to access from one domain IT resources located in another domain, with sufficient and appropriate access rights. This interoperability between domains will take effect at runtime, when live authorization decisions are taken, and in an out-of-band mode, in order to compare the meaning and objectives of the security policy of each domain.

MULTIPOL will provide an innovative, modular and consistent security suite to implement strong security features between independently administered domains. This set of modules will implement coherent authorization features, taking into account that all domains are enforcing a different security policy.

PREDYKOT (Policies REfined DYnamically and Kept On Track)
PREDYKOT provides an innovative, modular and consistent suite to ensure that a policy (access control security, SLA etc) remains efficient whatever changes occur to it: administrative, contextual etc.
Smart access control for online networks Putting intelligence into organizations'' security policies. Defending Internet, intranet and extranet connections against unauthorized access has become an ever-expanding challenge as use of the virtual world has grown.

The ITEA 2 PREDYKOT project has developed a new approach to security policies for organizations that will enable security strategies and software to formalize a decision process, dynamically respond to organizational changes, and reconfigure them to adapt to new conditions.

4. Attack types

To understand the attacks to assets of information technologies, first security threats must be defined. Security threats are grouped together into four;

- Disclosure
  - exposure, intercept, inference, intrusion
- Deception
  - masquerade, falsification, repudiation
- Disruption

- incapacitation, corruption, obstruction
- Usurpation
  - misappropriation, misuse

4.1 Physical mediums

Especially channels are exposed to security threats in wireless networks. *Maximal Interception Attack*, *Spectral Honey Pot*, *Jamming*, *Selfish Use* and *Common Control Channel Attack* are defined in next section. On the contrary, wired networks are rarely exposed to disclosure, deception or usurpation based threats. However, deception based attacks, namely corruption of the network equipment or obstruction of the network software, are possible threats. In addition, TEMPEST must be considered to the threats of disclosure of the wired networks.

4.2 Entities

Interference may be exploited to the users. Mask Primary User, Hidden Node are interference based threats to wireless network nodes. In addition, learning nodes in cognitive radio nodes may be threatened by being a virus. Wired or wireless networks may have some data to control the network. To obstruct the network functions falsification of these data may be exploited. Policy and belief Manipulation are some of these types of attacks.

4.3 Functions

Most of the functions are based on the network layer. Traffic Analysis, Looping, Detouring and Black/gray hole attacks are network based attacks. Transport layer is another layer has a lot of type of attacks. DoS attacks are based on every layer but most effective DoS attacks are mainly based on transport layer.

4.4 Complex attacks

A complex attack is different from the other attack types due to coming from collaborating attackers or an attacker with various identities or being produced in temporally and/or spatially sequences and stages. Although each action in itself may not violate any rules explicitly, entirely the total of the actions are a threat to the systems.

4.4.1 Examples of Complex Attacks

DDoS Attacks

DDoS attacks are exploited by multiple users to disrupt services in a system. Detailed explanation is given in this section.

Cross-layer attacks

Considering each layer in the protocol stack an attacker may exploit a threat based on more than one layer or attack to an individual layer to affect another layer. This type of attack is defined as cross-layer attack. Defense mechanisms based on unique layers are weak to cross layer attacks due to trusting the lower or upper layer communications. In addition, the networks aimed at increasing the efficiency through exchanging information among different layers, as in cognitive radio networks, employ cross-layer design and opens to cross-layer threats.

The Lion attack is defined in [8]. Physical layer disruption attack, jamming, is a forcing mechanism to cognitive radios implement frequency handoff. Continuously handoff may degrade the TCP throughput. The mitigation method for jamming can be a threat for another layer. Additional to traditional security mitigation methods, a cross-layer IDS is suggested in the same study.

Sub-optimal performance of network-aware spectrum division:

In the collaborative manner, attackers can change the network topology and affect the spectrum division efficiency to create minimum spectrum divisions. This type of attack is common in cognitive radio networks. Network aware spectrum division methods must consider the performance of the spectrum as well.

Sybil attacks

Collaborative applications are easily disrupted by uncooperative and malicious users. Those users profit from services without providing an adequate return and then make themselves untraceable by creating a very large number of bogus identities are identified as Sybil [39].

Especially dynamic spectrum access environment is exposed to complex attacks some of the attacks to DSA are Environment tampering, Reverse engineering Online prediction with expert, Insecurity in fast handover mechanisms, Multiplicity and indeterminacy problem.

4.4.2 Mitigation Methods to Complex Attacks

[40] and [41] suggest cross-layer designed IDSes for cognitive radio networks. Cross-layer designed IDSes are efficient at cross-layer attacks. In addition, the commercial IDS is aware and tries to find solution for complex attacks. Sophisticated signature identification methods are used to mitigate complex attacks. A patented technique Compound Signature Identification (CSI) is employed by Check Point and Juniper Networks at their state–of-the-art IPSes. CSI matches some pre-defined signatures consecutively and if a certain logical condition over the multiple contexts is matched, a response is done. CSI is an efficient mitigation method to temporally/spatially designed complex attacks [32]. Out-of-date CAPICOM (Cryptographic Application Programming Interface Component Object Model) had been a layered mechanism for the complex attacks. Three signatures were validated to take the appropriate action.

Signature based methods are easier than the anomaly based detection of complex attacks, due to anomaly based methods' requiring a learning mechanism to differentiate the normal behaviors of the host or the network that is under protection. Besides, anomalies detected need to be analyzed by trained human operatives. This makes an anomaly-based detection much more supervised solution than signature IDS.

Spatio-temporal outlier detection methods may be employed as a detection method to complex attacks.

Complex attacks are examined and known mitigation methods are explained in this document. Still the attackers may exploit some threats to the assets. An attacker may emulate a physical medium as control medium and capture the data flow on it, manipulate the policy on central authority and corrupt a physical medium permanently, intrude a dedicated network, emulate a network and attract the user to be contained in or exploit a Sybil to implement a malicious network, corrupt a network by misappropriating the central authority, intrude and exposure the useful information from network, emulate the information and falsify the knowledge bases. In addition, fast handover mechanisms and unwise planned frequency divisions, cross-layer approaches, routing mechanism, learning algorithms and online experts are open to threats. Repudiation is still open to be researched. Sybil attacks are based on repudiation mechanism and also form a starting point to the other attack types. As a result of these flaws, a unique identification system, feasible physical layer encryption, counter-attack methods must be investigated.

4.5 DDoS attack types

The denial of service attacks, distributed or not, can be executed in different ways and targeting different resources to achieve the effect of denying legitimate users the use of the service. First of all, we consider here only remote denial of service attacks, perpetrated over the network and leave out the local attacks, as the remote attacks are far more common thanks to their easy execution. In [44], the authors present a taxonomy for both DDoS attacks and defenses, which takes into account various aspects going from the scanning and recruiting of zombie machines for distributed attacks to attack dynamics. For the purposes of the project, we can narrow down the differentiation of DDoS attack to exploited weakness and the target type.

4.5.1 Exploited weaknesses

DDoS attacks can be separated in two categories, *brute force* and *semantic* attacks, by the type of the resource they are trying to consume at the target.

- *Brute force* attacks are based on the sheer volume of the traffic they generate. For example, a 1 Gbps link receiving more than 1 Gbps traffic is simply unable to handle the traffic volume causing packet drops for both attack and legitimate traffic – the exact dropping behavior depending on the networking equipment. On some target types, typically ones that do at least part of the processing in software, the brute force effect can be achieved before the link is saturated at layer 2, as the packet processing requires resources. For example, some of the authors discuss the packet processing impact on general purpose operating systems, that can be freeze by only packet reception. Brute force attacks are often easier to detect than other attack types due to the significant volume. The result can be achieved with any type of traffic the attacker is able to generate, directly or indirectly. As the only requirements are the volume and sufficient level of correctness that the traffic reaches targeted components (typically meaning correct layer 3 packets so that the routers forward it to the destination), “simple” flooding using ICMP, UDP and/or TCP packets can be used. For the attacker aiming to saturate the available bandwidth, it can be advantageous to generate large packets so that the same effect would be achieved with lesser number of packets. The attacker can also seek to amplify these types of attacks by using so called reflectors.

- *Semantic attacks* exploiting a weakness in protocol design or implementation to consume resources on the target. Many of today’s protocols have design issues that allow perfectly correct packets be used cause denial service conditions with smaller amount of traffic than by brute force attacks. Maybe the most classic example is SYN flooding, which can fill up the operating system’s state tables by bogus connection requests that are newer completed, and deny legitimate clients from establishing new TCP connections. Semantic attacks can also target applications, again taking advantage protocol design that allows tying up application resources or by using the application behavior to send requests that consume a lot of resources like CPU, memory, disk I/O and/or network bandwidth. For example, Slowloris is a tool targeting some web servers, by starting to send several, protocol-wired legitimate, HTTP requests that it never completes, but maintains alive with a small amount of packets. The web server can only serve a limited amount of connections, and once the tool once the tool occupies all of them, no new connections can be handled by the web server. Another example is repeated requesting of a heavy resource that requires CPU cycles (e.g. cryptographic operations related to authentication) and/or bandwidth (e.g. large images), which can degrade the server performance.

4.5.2 Target type

We differentiate three categories of targets: *network*, *host*, and *application*. There is often a link between the exploited weakness and the attack target, and different targets cause different amounts of collateral damage.

- Attacks targeting a *network* aim at saturating the network links or the networking equipment like switches, routers, load balancer, and/or firewalls somewhere on the attack path. Typically attacks targeting the network are brute force attacks or semantic attacks against state tables in intermediate equipment like firewalls or load balancers. If these attacks are successful, everything that is behind the targeted equipment becomes unavailable. For example saturating the edge router(s) or the Internet connection(s) of a site renders the whole site unavailable (and unable to communicate towards exterior as well).

- Attacks targeting the *host* aim at saturating host resources, typically the TCP state tables or the packet processing capacity of the end host. Typically attacks targeting a host are either brute force in terms of number of packets or semantic attacks. A successful attack renders all services depending on the underlying networking stack inoperable. For instance, a SYN flood that saturates the state tables cripple all applications running on TCP.
- Attacks targeting an *application* aim at saturating its resources and exploit application or application protocol semantics. A successful attack can impact only the application itself, e.g. in the case of Slowloris, another web server running on the same host would continue to function normally. Attacks targeting applications that run on TCP have the particularity that they more or less prevent the use of spoofed address, as the TCP connection must be established before the attacker can interact with the application.

5. Intrusion detection methods and systems

5.1 Intrusion detection methods

In this section, we provide a literature review of Intrusion Detection Systems (IDSes) along with performance measures.

IDSes are categorized into three main types according to their methodologies: *anomaly*, *signature* (misuse) and *signature-inspired* detection techniques. Misuse detection techniques first define the character of the attacks, and then a different type of actions, which are not defined in attack character, is categorized as normal usage. In contrary, anomaly based detection techniques first define the characteristic of normal usage, and then any deviation from normal character is determined as an anomaly (attack). Signature inspired techniques consist of both detecting attacks defined in signature databases and searching for anomalies in the networks. A literature review of signature detection and anomaly detection techniques is provided below.

5.1.1 Misuse detection based techniques

Misuse detection based algorithms are most widespread techniques used in commercial IDS products. Monitored information collected from different types of resources is compared with the defined attack patterns. Accordingly, in this technique it is possible to detect only the attacks, which have similar characteristics to defined attacks [1]. Therefore, the success rate of misuse detection based IDSs depends on how much information is preprocessed into detection engine of IDSs [2]. Misuse detection based techniques can be categorized as signature based, rule based, state transition based and data mining based algorithms.

Signature based algorithms can be carried out easily. For this reason, these algorithms are very popular in commercial products. In signature-based algorithms, attack signatures are determined according to the semantic characteristics of attacks. An ID compares monitoring data against attack signature database to decide if it is an attack or not [3]. Several signature matching algorithms are used in various signature based IDSs such as Snort [4], Haystack [5], NFR [6], Bro [7] and ARMD [8].

Rule based algorithms are based on several conditional rules. Monitoring data are compared with these rules to detect an attack. IDES [9], NIDES [10], MIDAS [11], EMERALD [12] and Computer-Watch [13] are some of the IDSs that use rule based algorithms. State transition based algorithms are based on finite state machines. Attacks are series of activities where each activity results in state transitions in a finite state machine. STATL [14], STAT [15], USTAT [16] and NetSTAT [17] are some of the IDSs based on state transition algorithms.

Data mining based algorithms categorize system usages as acceptable or unacceptable. By using data mining algorithms, historical data of monitoring system are analyzed. MADAM ID is a data mining based algorithm, which uses several classification techniques to detect an attack [18].

5.1.2 Anomaly detection based techniques

Anomaly detection based techniques rely on the idea of determining the normal behavior and then detecting the attack by analyzing the deviation from normal behavior. This technique is also resistant to novel attacks, since the system does not depend on the features of known attacks. However, in this technique the most important challenge is to determine the distinction between normal behavior and attack behavior. Anomaly detection based techniques can be categorized as *statistical modeling*, *classification*, *clustering* and *neighboring* based approaches.
In statistical modeling approach, acceptable behavior is determined according to system usage policies. IDS detects if there is any deviation from normal usage threshold with the help of several statistical techniques. If any monitored data exceed the threshold, this anomaly is called attack. Several statistical modeling techniques are used such as Hidden Markov Model in IDSs [19]. In addition, some misuse detection based IDSs such as IDES [9], NIDES [10] and EMERALD [12] have also a statistical component, which is used for anomaly detection.

Artificial Neural Networks are utilized to detect new attacks by using supervised learning methods [20]. Also Fox et. al. proposed another work, which is based on neural networks using unsupervised learning methods [21]. There is also another work, which is called Neuro-Fuzzy IDS (NFIDS), which uses both neural networks and fuzzy logic [22]. Abouzakhar et al. [23] also proposed an IDS for detecting distributed DoS attacks by using Neuro-fuzzy techniques. Yao et al. [24] used a hybrid intrusion detection system, which is based on fuzzy logic and Support Vector Machine techniques. Bayesian networks are also utilized in IDS algorithms [25]. Classification based methods have merits of performing very fast at online data and disadvantages of the requirement of a supervisor or a training period for the classifier. In [1] classification based intrusion detection methods are examined.

Clustering based methods with the framework of the unsupervised learning mechanism are applied despite not being convenient to the online data. Hierarchical, partial, dense, grid based and fuzzy clustering methods are employed to detect anomalous observations. Local Outlier Factor [26], K-means algorithms [28] are based on clustering. The deal with the number of clusters is the main concern in this method.

Outlier based techniques are also utilized in IDSs. If a data point is very far from other points, it is called outlier. In other words, if data point does not fall into a range according to a standard deviation and mean, then this point is an outlier. Similarly, in IDSs attack is considered as an outlier, since it is different from normal acceptable usage.

Neighboring based methods are based on a similarity or a distance function. The observations very far from the general are outliers. Nearest neighbor algorithm [29] is used in IDSs.

5.1.2.1 Bayesian multiple change point model for detection of abrupt changes

Statistical Machine Learning (ML) is about detecting structural patterns in multivariate data by the use of probabilistic models and associated inference algorithms. Rather than describing step-by-step an algorithm to arrive at a desired solution, the common paradigm is describing a phenomenon with the data collected from the process of interest, typically as a collection of features. The structure of the features is described by probabilistic models and statistical inference provides a means of deriving algorithms for arriving at the desired solutions. In this project, intrusion detection approaches will be investigated that employ two different machine learning approaches: classification and clustering.

When the true categories of data instances are known, this information can be used to detect the similarities of the data features. This is called a supervised method, *classification*. Once those similarities are learned, the category of a new, not previously observed data instance can be inferred. If we have no knowledge for the true categories of data instances, we can still search for similarities together with possible category assignments. This is called an unsupervised method, *clustering*.

In recent years techniques form Bayesian statistics became very popular in diverse fields such as machine learning, bioinformatics, finance, and signal processing. The common aspect in all these applications is the presence of noisy data and the uncertainty regarding the underlying data-generating process. Moreover, plenty of expert knowledge is available; however, it is not clear how to incorporate this into a rigorous statistical framework. Bayesian techniques offer an elegant solution to this problem by the use of probabilistic models in a general and well-defined computational framework. A Bayesian model consists of two components:

- A prior distribution summarizing expert knowledge in terms of unobserved parameters,
- A likelihood component, which describes the conditional probability of observed data given a particular setting of parameters.

By calculating the posterior probabilities of the parameters, one can infer desired information about the data generation process as well as carry out model comparison and selection. In the Bayesian approach, a model with a set of parameters is proposed to explain the behavior of the data.

Bayesian Inference

In Bayesian statistics, probability models are viewed as data structures that represent a model builder's knowledge about a (possibly uncertain) phenomenon. The central quantity is a joint probability distribution:

$$p(y, x, \theta) \quad = \quad p(y|\theta, x)p(x, \theta) \tag{5.1}$$

that relates unknown variables x and unknown parameters θ to observations y. In probabilistic modeling, there is no fundamental difference between unknown variables and unknown model parameters; all can be viewed as unknown quantities to be estimated.

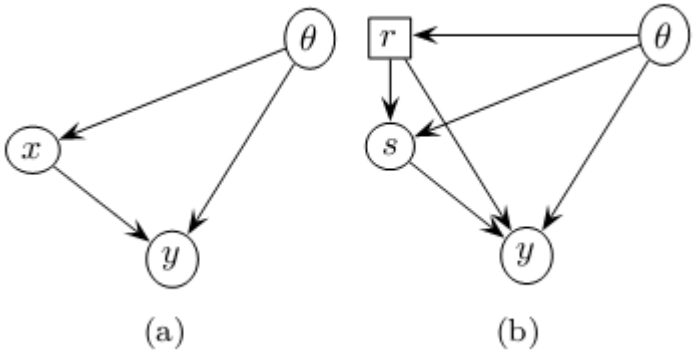


Figure 5-1 Directed graphical model showing the assumed causal relationship between observables y, hidden x and parameters θ. (b) The hidden variables are further partitioned as x= (s;r). Square nodes denote discrete, oval nodes denote continuous variables.

The inference problem is to compute the posterior distribution using the Bayes theorem:

$$p(x, \theta|y) \quad = \quad \frac{1}{p(y)} p(y|\theta, x)p(x, \theta) \tag{5.2}$$

The prior term p(x;θ) reflects our knowledge about the parameters θ and hidden variables x before we observe any data. The likelihood model p(y/θ;x) relates θ and x to the observations y. It is usually convenient to think of p(y/θ;x) as a generative model for y. The model can be represented as a graphical model shown in Figure 5.1. Given the observations y, the posterior p(x;θ/y) reflects our entire knowledge (e.g., the probable values and the associated uncertainties) about the unknown quantities. A posterior distribution on the hidden variables can be obtained by integrating the joint posterior over the parameters, i.e.

$$p(x|y) = \int d\theta p(x, \theta|y)$$

(5.3)

From this quantity, we can obtain the most probable x given y as x

$$x^* = \arg\max xp(x|y)$$

(5.4)

Unfortunately, the required integrations are in most cases intractable so one has to reside to numerical or analytical approximation techniques. At this point, it is often more convenient to distinguish between x and to simplify approximations. For example, one common approach to approximation is to use a point estimate of the parameter and to convert intractable integration to a simple function evaluation. Such an estimate is the maximum a-posteriori (MAP) estimate given as:

$$\theta^* = \arg\max \theta \int dx p(x, \theta|y)$$

(5.5)

$$p(x|y) \approx p(x, \theta^*|y)$$

(5.6)

Note that this formulation is equivalent to “learning" the best parameters given the observations. In some special cases, the required integrations over θ may still be carried out exactly. This includes the cases when y, x and θ are jointly Gaussian, or when both x and θ are discrete. Here, exact calculation hinges whether it is possible to represent the posterior p(x; θ/y) in a factorized form using a data structure such as the junction tree.

Bayesian Networks

Bayesian networks (BNs), also called *belief networks*, *Bayesian belief networks*, *Bayes nets*, and sometimes *causal probabilistic networks*, are an increasingly popular method for modeling uncertain and complex domains. A Bayesian network models a joint probability distribution in the following factorized form:

$$p(x_1, x_2, \ldots, x_n) = \prod_{i=1}^{n} p(x_i|pa(x_i))$$

(5.7)

where pa(x) denotes the 'parents' of the random variable x_i. One can associate with a given factorization a directed acyclic graph structure where each node corresponds to a single random variable x_i and pointed by the parent nodes. An example is given in Figure 5.2. The underlying graph is useful for both modeling the domain and carrying out efficient inference via message passing algorithms.

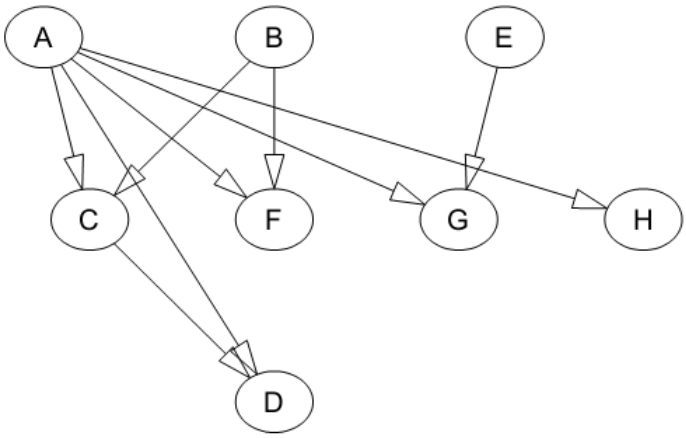


Figure 5-2 A Bayesian network.

More precisely, we depict a Bayesian network using a graph in which a node represents a variable Xi and the variables which point to xi are the parents of this variable. Each node in the network then corresponds to a factor in the joint distribution over all variables, see Fig. 5.2. This model is encoding the following joint distribution: p(A)p(B)p(E)p(C/A;B)p(F/A;B)p(G/A;E)p(H/A)p(D/C;A).

By Bayes' recursive construction, the graph must be acyclic. The most general form of a Bayesian network is therefore the cascade graph in which the parents of a variable are all the previous variables in the ordering. Any valid network can be obtained by removing edges in the cascade graph, with each removal corresponding to a conditional independence assumption.
Graphs have a long history in the description of probability models and it is important to stress the difference between a probabilistic graphical model and alternative graph representations such as state transition diagrams or block diagrams that use an entirely different set of semantic rules. More generally, probabilistic graphical models are compact depictions of independence and factorization assumptions of a probability density. Besides the directed acyclic graphs, there exist also other formalisms. Two well-known are *undirected models* and *factor graphs*.

The Bayesian networks methodology emerged first in the artificial intelligence research and adopted quickly by the statistics communities. It has been applied to a wide range of problems, ranging from text analysis, medical diagnoses and the evaluation of scientific evidence. Bayesian modeling techniques have several features that make them useful in many real life data analysis and management questions. As one works with probabilities, the methodology provides a natural way to handle missing data, and allow combination of data with prior domain knowledge. Furthermore, Bayesian networks provide a method for avoiding over fitting of data, they can show good prediction accuracy even with rather small sample sizes, and they can be easily combined with decision analytic tools to aid management, as such they are also a useful tool for expert elicitation and combining uncertain knowledge when used with care.

Advantages of Bayesian Networks

- *Suitable for small and incomplete data sets:* There are no minimum sample sizes required to perform the analysis, and BNs take into account all the data there is.
- *Structural learning possible:* In addition to defining the model structure based on subject matter knowledge and using the data to define the conditional probability distributions, it is also possible to use data to learn also the structure of BN.
- *Combining different sources of knowledge:* An important feature of Bayesian methods is the use of prior information.
- *Explicit treatment of uncertainty and support for decision analysis:* Bayesian networks can easily be supplemented with variables encoding managerial decisions that in their turn affect the natural variables of the model, and with variables encoding costs and utilities related to these decisions and their outcomes.
- *Fast responses:* Because BNs are solved analytically; they can provide fast responses to queries once the model is compiled.

Disadvantages of Bayesian networks

- Whilst modeling continuous variables are possible, exact inference in mixed models (containing both discrete and continuous variables) can become quickly intractable, requiring approximate inference techniques relaying on mean field or Monte Carlo methods. In complex models, there may not be always exact guarantees for the computed approximations.
- *Discretization of continuous variables:* Data and parameters often have continuous values.
- Bayesian networks can, however, deal with continuous variables in only a limited manner.
- The usual solution is to discretize the variables and build the model over the discrete domain.
- *Collecting and structuring expert knowledge:* While Bayesian models are a useful way to model expert knowledge; it may prove difficult to get the knowledge out of the experts in a form that can be converted into probability distributions.

Evaluation Criteria and Performance Metrics

The first step in the evaluation of the classification algorithms is to find the true classes (the ground truth) of the applications by inspecting the network packets. The overall performance of classification algorithms (classifiers) are usually evaluated according to their accuracy. The accuracy is defined as the percentage of the correctly classified instances among all instances. This metric can be used to describe the success of the whole system and also can be used as a per-class success metric. In addition, other per-class measures assess the success of a classier for a given class. The simplest per-class metrics are given by:

- *False Negatives (FN):* Percentage of members of class X incorrectly classified as not belonging to class X.
- *False Positives (FP):* Percentage of members of other classes incorrectly classified as belonging to class X.
- *True Negatives(TN):* Percentage of members of class X correctly classified as belonging to class X (100%-FN).
- *True Positives (TP):* Percentage of members of other classes correctly classified as not belonging to class X (100%-TN).

Additionally, *precision* and *recall* are two per-class metrics written in terms of true and false positives and negatives. Precision is the percentage of instances that are properly classified in a class, and recall is the percentage of instances in a class that are properly identified.

5.2 Intrusion detection systems

The intrusion detection concept was suggested by James Anderson [29] with the aim of helping the security officers. Now there are numerous commercial and open source intrusion detection systems and some are equipped with a self-defense mechanism. The IDSes help security systems protect against theft of data, financial loss, persistent threats, targeted attacks, and DOS attacks. There are many companies that produce IDS systems but in the context of technology, product roadmap, security, management, price-to-performance ratio, pricing, financial stability, and service and support, seven producers are coming into prominence [30]. CheckPoint, Cisco, Enterasys, HP/TippingPoint, IBM ISS, Juniper, McAfee, SonicWALL, Sourcefire, and Symantec are featured companies.

Open source IDS systems are fully or partially employed for detecting and preventing against the threats. Snort, OSSEC HIDS, OSSIM and Honeyd are widely used [31], Suricata being a more recent open source IDS.

5.2.1 Commercial Intrusion Detection Systems

Check Point IPS Software Blade detects the attacks by misuse and anomaly detection methods. Both methods are based on rule-based classifier. DoS mitigation engine protects against DoS, and DDoS attacks. IPS Performance is stated to be up to 15 Gbps. Rule based protection is updated for client and server vulnerabilities, exploits, protocol misuse, outbound malware communications, tunneling attempts, application control, generic attack types without predefined signatures and preemptive security functions. According to reports, the signature quality is consistently low [32].

Cisco IPS sensors have a statistical based detection engine and has a four Gbps throughput. Meta Event Generator (MEG), a threshold and rating based algorithm, is employed to detect the DoS attacks. TCP-reset, packet drop and firewall reconfiguration are mitigation methods for detected attacks. The logging service is file based. Since Cisco is a dominant network infrastructure vendor, Cisco IPS easily manages the other network devices. Management console is not scoring well against the other products [30]. Besides, an inexperienced administrator can reduce the protection mechanism by setting the risk-rating feature unintentionally.

Enterasys Intrusion Prevention System employs a rule-based method for detection up to ten Gbps throughput. Limiting the inbound by pre-defined rates help degrade the DDoS attacks. TCP-reset, ICMP unreachable and user-defined actions are mitigation methods for detected attacks [33]. However, Enterasys' signature library is too large to have a high fidelity. In addition, the blocking mode is not working well [30].

HP/TippingPoint employs a statistical based detection engine and has an up to 16 Gbps throughput. Traffic blocking is a mitigation method for detected attacks [34]. Security effectiveness has some critics [30]. IBM Proventia series has a main advantage with its in-line simulation to determine what the best blocking behavior is before activating the blocking [35]. 1.2 Gbps throughput is a disadvantage of Proventia. Juniper Netscreen detects the attacks by a basic signature-matching algorithm. Logging is based on database or a file system and TCP-Reset, ICMP-Unreachable are mitigation methods to detected threats. 30 Gbps throughput is a distinctive feature of Netscreen. IDPS security effectiveness is criticized by most tests [30]. McAfee Intrushield Statistical based method detects protocol and application anomalies and DoS attacks. TCP reset, ICMP unreachable, dropping of packets, firewall reconfiguration are mitigation methods. Ten Gbps throughput is on average in the market.

Sourcefire is the commercial manager of the Snort and Clamav. Next-Generation IPS of Sourcefire has an open-source Snort engine. Although IPS has 20Gbps throughput, lack of firewall is a disadvantage. Emerald is a rule-based IDS and also has a Bayesian inference system. The distributed detection and response system performs statistical profile-based anomaly detection.

5.2.2 Open-source Intrusion Detection Systems

Open platform Snort is rule-based system and has the multiple anomaly-based third party modules. TCP reset, ICMP unreachable, configuration of firewalls, alerting via email, pager, and SMS (plugins) are possible reactions. Actions are reported into log files, log server or a database. There is also a detection algorithm to DoS attacks [36].

OSSEC HIDS [37] is an open source host-based intrusion detection system based on signatures, and performs log analysis, integrity checking, root kit detection, time-based alerting, and active response. OSSIM has a MAC address and service anomaly detection and also can employ Snort as a facilitator.

Bro IDS is a signature based intrusion detection system. The difference among the Bro and the rest IPS systems is using a scripting language, especially designed to facilitate network-traffic analysis and to detect anomalies. To reset the connection, reconfiguring firewalls are main mitigation methods to threats.

5.3 DDoS Attack Detection and Performance Limits

Detection of DDoS attacks accurately is not a trivial issue because of two following reasons[56]:

- Attack methods and behavior are developing very fast that they become sophisticated and complicated at short notice. Eliminating attack behavior from normal traffic becomes a major challenge for detection of DDoSes.
- Due to the fast enormous data in current network, detecting attacks quickly is significant.

By considering these challenges, several methods are proposed to detect DDoS attacks.

5.3.1 An approach to SYN flood detection

One of the well-known DDoS attacks is TCP SYN flooding attack. This attack exploits three-way handshake protocol of TCP. In this protocol when the server gets the SYN request message, it sends back SYNACK packet and then wait for the client to send final ACK message.

There is a half-open connection while waiting for final ACK. Since attacker uses a spoofed address, server cannot get the final message, there will be a constant half-open connection. Since server has limited resource, it cannot serve for oncoming connection. In this situation, early detection is important, since it will give enough time to prevent attacker exploiting all the half-open connections. Passive detection methods cannot manage to detect in early stages, since they rely on passively sniffing an attacking signature. For these reason an active detecting approach is suggested in [58]. A delay probing method DARB is proposed in this paper. In active detection approach, first step is to determine the reason of half-open connection. Normal half-open connections are caused by congestion, but if there is no sign of congestion, then delay between server and client is abnormal and there may be an attack. By using delay, probing method (DARB) delay between server and client is estimated. Different time to live (TTL) values in IP headers of packets will result in death of packets in different routers. These routers send information about death time of packets. The delay can be estimated by the help of this information. Then a score is given to delay value which gives the probability of the reason of half-open connection is a SYN attack. An experiment is designed if this method can distinguish normal half-open and abnormal half-open connections to detect SYN flooding attack.

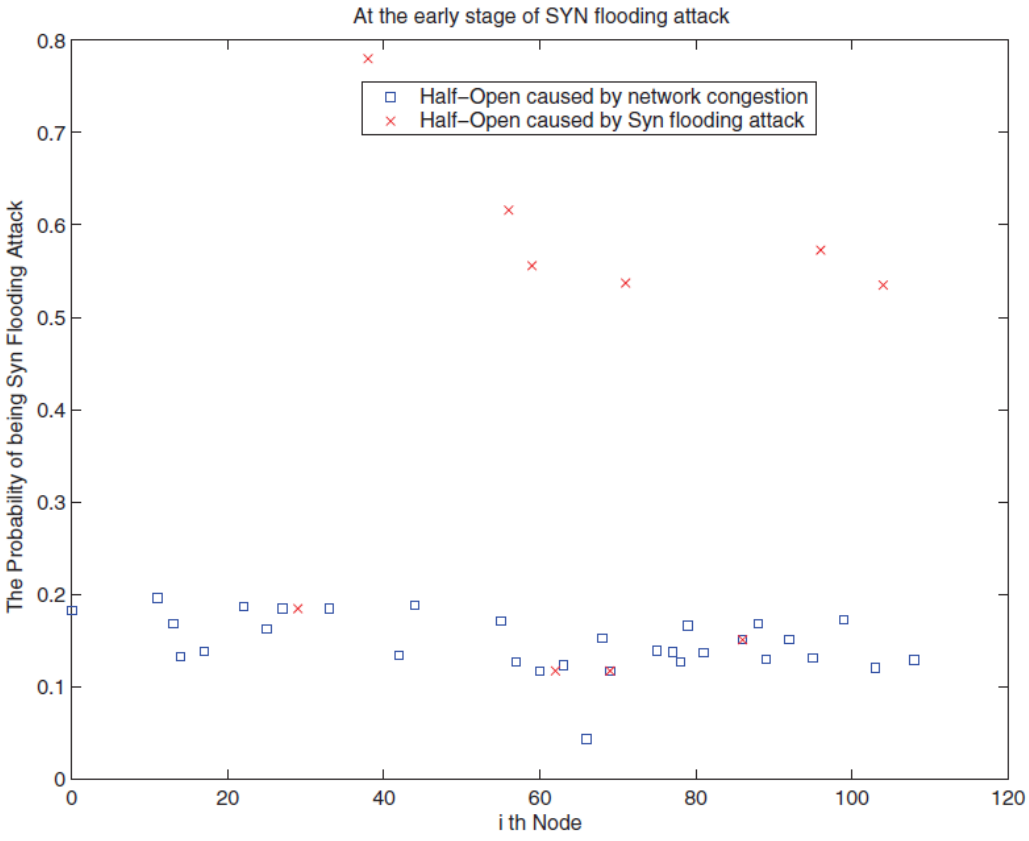
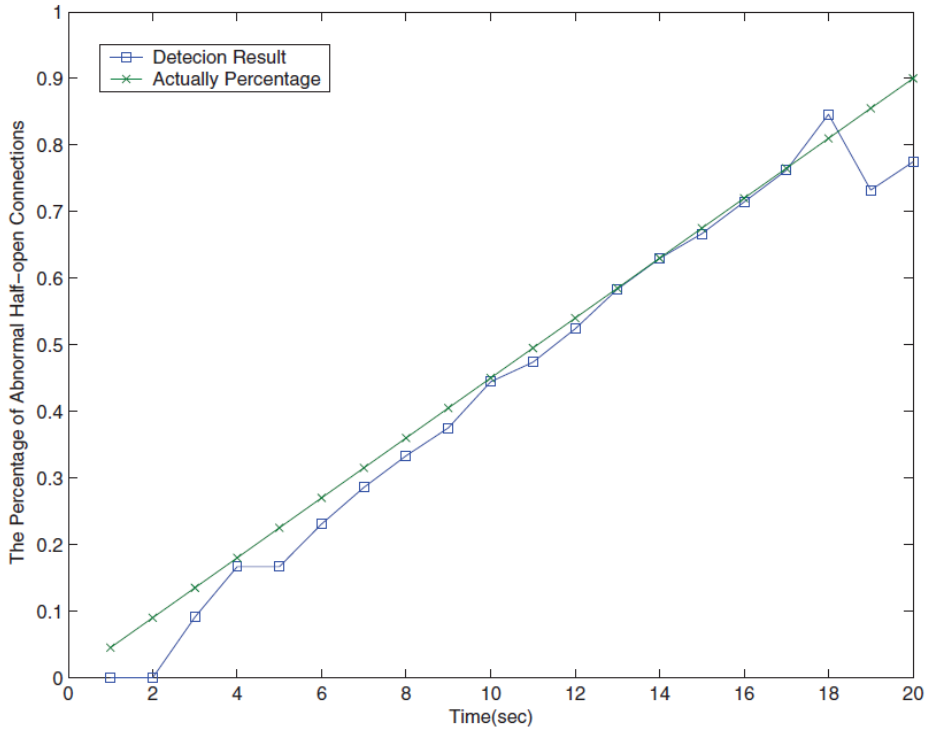
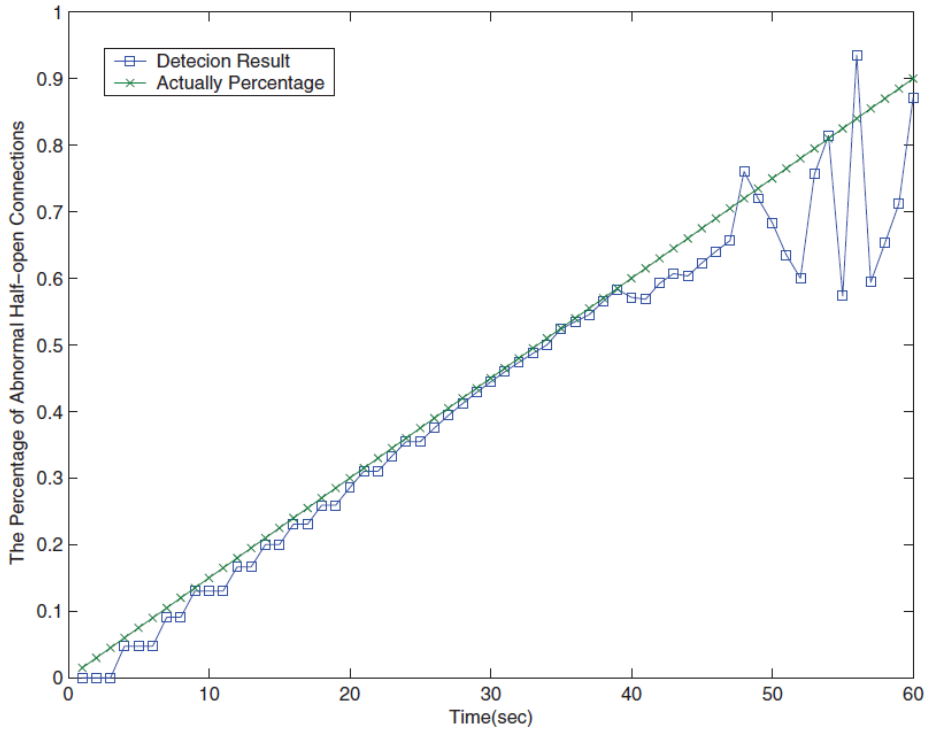


Figure 5-3 The probability of half-open connections indicates a SYN flooding attack

In the Figure 5.3, it is easy to classify normal and abnormal half-open connections since abnormal half-opens have much higher score than normal half-opens. For this experiment, if SYN-flooding attack probability is more than 0.5, it is abnormal half-open connection. Another experiment is also designed to test the detection rate by various SYN-flooding attack rates in 20 and 60 seconds.



(a) Within 20 sec



(b) Within 60 sec

Figure 5-4 Detection of SYN flooding attacks at various rates

Figure 5.4 shows that detection results are accurate in early stages (20 seconds). However, there are some vibrations in later stages (60 seconds) since they use a sampling method in probing which affects the accuracy. According to these results, suggested method works well in early stages, whereas it is more erroneous for later stages of the SYN-flooding attack.

5.3.2 An approach to general flood detection

In another paper [56], a system that only extracts several important attributes from network traffic to detect several types of DDoS attacks such as ICMP flooding, TCP flooding, TCP-SYN flooding, UDP flooding, and Smurf style attacks. Many of the existing detection methods use 41 attributes, which are defined in [18]. However, some of the attributes are useless or noise which decreases the accuracy and performance of the system. By utilizing Information Gain and Chi-square methods, importance of these attributes are ranked. Bayesian network and decision tree (C4.5) is used to detect DDoS attacks and decide how many attributes are enough for detection. Detection Rate (DR- percentage of intrusions detected) and False Positive Rates (FPR-percentage of normal connections falsely classified as intrusion) are utilized for evaluation criteria.

No. of attributes	C4.5		BN	
	DR(%)	FPR(%)	DR(%)	FPR(%)
1	99.3	0.4	99.3	0.4
2	99.4	0.5	99.2	0.2
3	99.5	0.7	99.3	0.2
4	99.7	1.5	99.3	0.4
5	99.8	2.3	99.1	0.2
6	99.8	2.3	99.5	0.4
7	99.8	2.4	99.3	1.5
8	99.8	1.1	99.5	2.3
9	**99.8**	**0.3**	**99.6**	**1.6**
10	99.8	0.3	99.5	1.6
11	99.8	0.3	99.2	1.5
18	99.8	0.3	99	1.5
27	99.8	0.3	99	1.5
41	**99.8**	**0.3**	**99**	**1.5**

Figure 5-5 DDoS attack detection results based on different attribute sizes using Bayesian networks and C4.5

According to Figure 5.5 , using only one attribute both Bayesian and C4.5 methods achieve high detection rate(% 99.3) and low false positive rate(% 0.4).Detection results remain same after 9 attributes for C4.5 method. In addition, for Bayesian method, detection rate starts to decrease which means that some of the attributes are irrelevant. Then smaller number of attributes will be enough to detect the intrusion.

No. of connections	Time used (s)	
	9 attributes	41 attributes
30,000	**237**	2043

Figure 5-6 Time used during attributes constructed

Methods	9 attributes		41 attributes	
	Training(s)	Test(s)	Training(s)	Test(s)
BN	**0.7**	**0.2**	4.4	0.9
C4.5	**1.7**	**0.2**	15.3	0.9

Figure 5-7 Time used during training and detection

Figure 5.7 shows that constructing smaller size of attributes, takes much shorter time. Construction of 41 attributes takes approximately 10 times longer time than constructing nine attributes. Similarly, it shows that smaller size attributes are much faster. As a result, nine attributes is sufficient to detect DDoS attacks effectively.

5.3.3 An approach to distributed detection

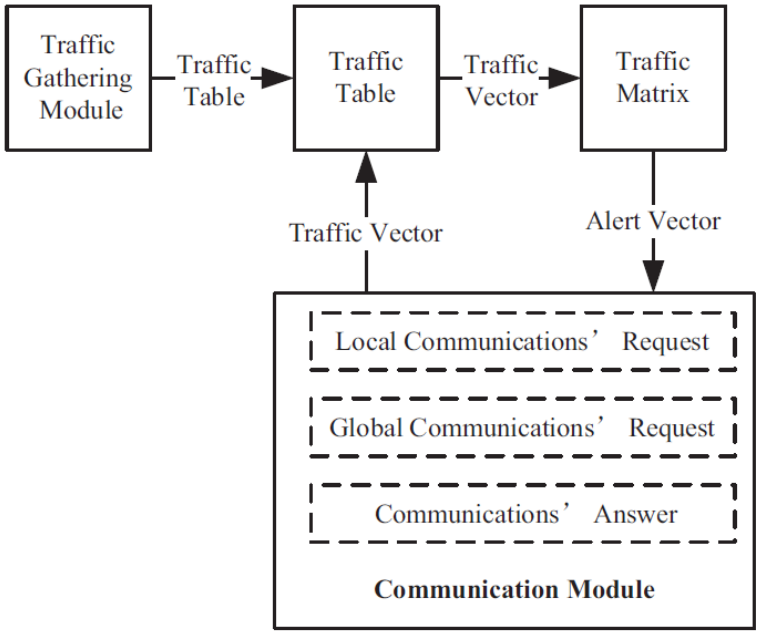


Figure 5-8 Organization of F-IDS

There is also a more recent work, which proposes a Distributed Intrusion Detection System called *F-DIDS* to detect flooding DDoS attacks [59]. F-DIDS gathers data with distributed components to analyze data from whole network. A single component in a specific network element is called F-IDS, which is based on traffic matrices. Each F-IDS communicates with other, could be the central processor and makes decision by itself. Since F-DIDS has a fully distributed architecture, there is a huge communication overhead.

This overhead is reduced by the proposed local and global communication methods. Organization of F-IDS is given in Figure 5.8. Traffic gathering module gathers the traffic information to store it into traffic table to enter data into traffic matrix. According to data in traffic matrix, alerts of DDoS attacks are classified into four levels (0,1,2, and 3). There is also an Alert Vector (AV) where AV = fFlood; SY N; ICMP; UDPg ; Flood; SY N; ICMP; UDP 2 f0; 1; 2; 3g. Flood is the possibility of TCP flooding attack. Similarly, SYN is SYN flooding attack possibility, ICMP is ICMP flooding attack possibility and UDP is UDP flooding attack possibility. Traffic communication (TC) packets are also used to transmit traffic data. There are three types of TC packets: local communications' request, global communications' request and communications' answer. An example of local communication is illustrated in Figure 5.9. F-IDS1 sends request TC to F-IDS2 and F-IDS3. Similarly global communication procedure is given in Figure 5.10. F-IDS1 sends global communication request. During global communication traffic vectors spread in the network. Flow chart of an F-IDS is also summarized in Figure 5.11.

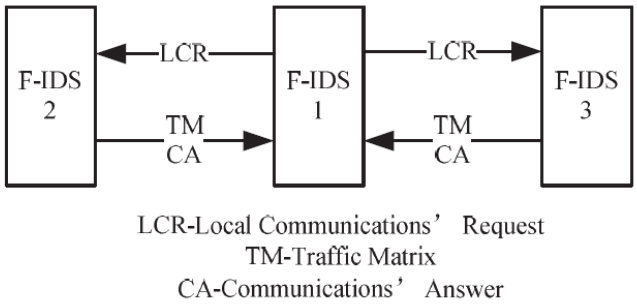


Figure 5-9 Procedure of local communication

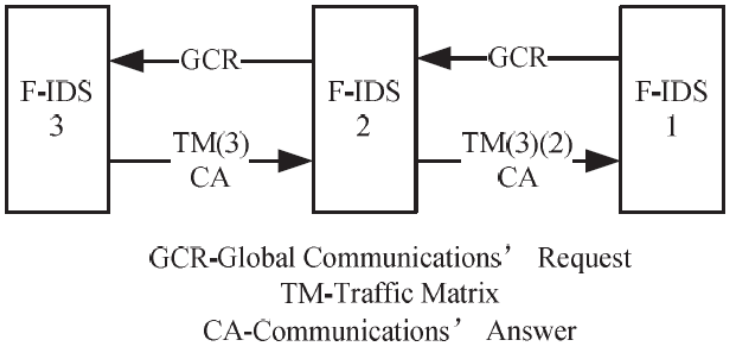


Figure 5-10 Procedure of global communication

They have simulated an F-DIDS with five nodes in NS-2 as illustrated in Figure 5.12. Suppose that Node0 is the victim. In addition, DDoS hacker is sending TCP packets through Node1, Node2 and Node3. The Figure 5.13 shows the incoming packets. In addition, Figure 5.14 shows the TCP flood level of alert vector in F-IDS0. From third to fifth second, alarm warning is given.

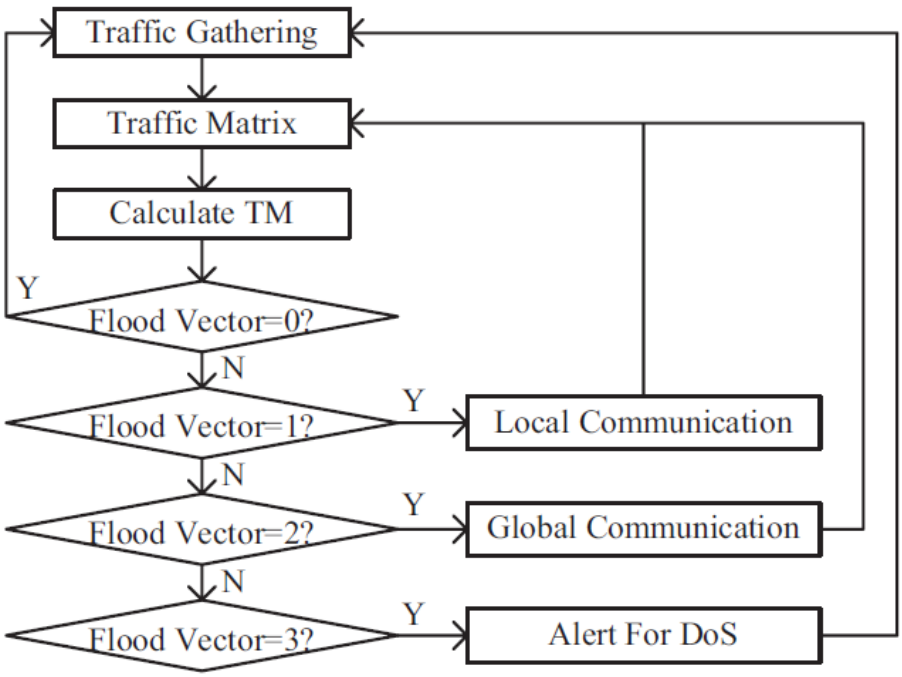


Figure 5-11 Flow chart of an F-IDS

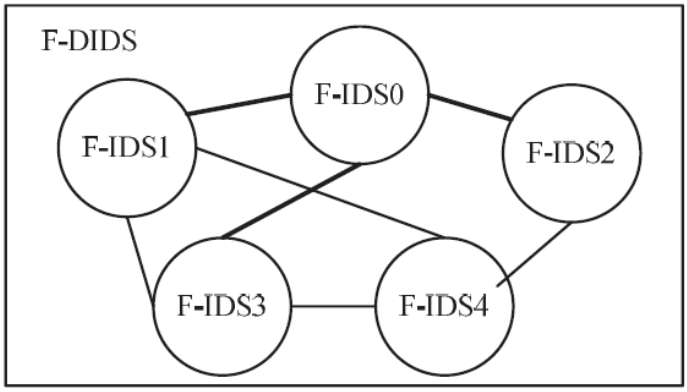


Figure 5-12 Topology of simulated network

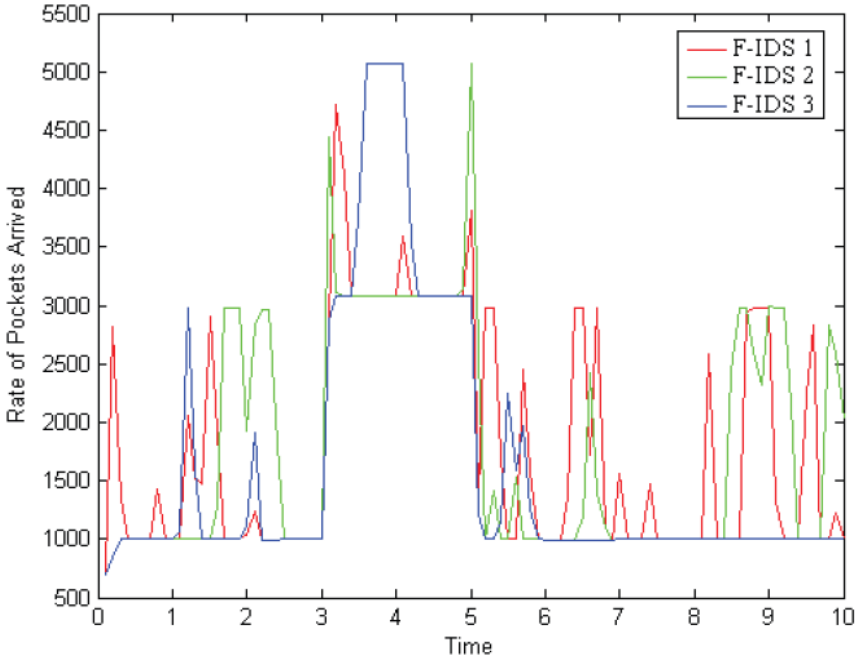


Figure 5-13 Incoming packets in three F-IDS

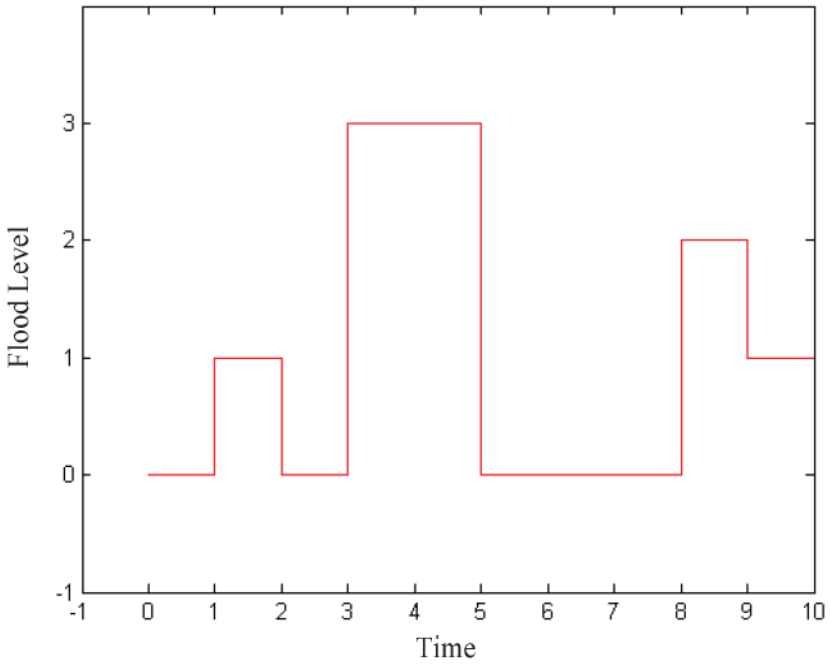


Figure 5-14 TCP flFlood level for F-IDS0

6. Intrusion prevention and reaction

6.1 Mitigation methods to cognitive based DSA

Dynamic Spectrum Access (DSA) provides radios with allocated spectrum in a productive and unbiased fashion. Channels are assigned dynamically by applying pre-programmed logic (adaptive radio), changing the transmission parameters manually (configurable radio), carrying out the rules (policy based), and adapting to the conditions by learning algorithms (cognitive radio). Cognitive Radio based DSA (CRbDSA) differs from the above mentioned ones. CRbDSA requires or reveals new assets to learn from and adapt to radio environment and share spectrum dynamically. Additional resources and stages are supplemented to CRbDSA to improve network and application performance, and form a collaborative network. Particular characteristics reveal new classes of security threats and challenges effective on specific domains of CRbDSA, and demand new security designs and implementations. Furthermore, Complex Attacks must be modeled in the context of different layers of communication framework of CRbDSA and stages of the Cognition Cycle. On the purpose of unveiling the security aspects; resources, communication layer properties and domains of CRbDSA, stages of the Cognition Cycle are explained in this section.

Resources:

Resources are particular entities equipped with realizing capability or affecting the Cognition Cycle and spectrum management. Primary and secondary users, primary and secondary base stations, and knobs are main resources of CRbDSA. Primary users and base stations are authenticated possessors of the licensed spectrum bands. Secondary users and base stations can occupy the vacant channels. Secondary users and base stations are based on a fully programmable software defined radio that should have policy, rule, optimization, and learning engines, geo-locator, sensors, knowledge base and any assistant sensing the environment. Agent based approach, game theory and machine learning are some of the disciplines assist CRbDSA.

Knobs are any entity perceived by the sensors of the users and base stations. For example, size and rate of the packets; data rate and frame type; noise power and magnitude are some of the network, link, and physical layer knobs respectively.

Stages of the Cognition Cycle:

Stages of the Cognition Cycle are defined by Mitola in [99].
Knobs are received from the environment in the Observe stage. The importance of the information gathered from the knobs is evaluated and determined, and a priority order is established in the Orient stage, By using new operating state information, alternatives are generated and determined in the Plan stage, In the Decide stage, an alternative is chosen in a way that presumably would improve the appraisal, In the Act stage, the resources are adjusted and allocated, messages are decided and sent, and signaling is performed, Finally, to improve effectiveness, the available actions are estimated in the Learn stage.

Stages of the Cognition Cycle are also defined as spectrum sensing and analysis (observe; orient), spectrum management and handoff (plan, decide, learn), and spectrum al-location and sharing (act).

Communication Layers:

CRbDSA provides robust, collaborative network and improved application framework by adapting Cross-layer design. Spectrum is a physical layer property and allocated into Channels. Each channel is employed as Data or Control Channel. Control channel performs negotiations on the other hand; data channel transmits the actual data among users and base stations.
Cognitive Radio Network (CRN) and Cognitive Network (CN) are designed to facilitate processing of the Cognition Cycle. Cognition Cycle stages receive the required knobs from mentioned networks. Cognitive radio network is formed to share physical, link and network layer knobs and information. On the other hand, Cognitive network shares the knobs of all of the communication layers.

Domains:

Defense in depth strategy is not just a military issue anymore. Multi-layered protection mechanism is formed to delay the attacker and yield additional time and space to defender. Other than communication layers and the Cognition Cycle stages, influential area of threat and countermeasure method may be used to defense in depth. These areas are defined as Domains of CRbDSA in this study.

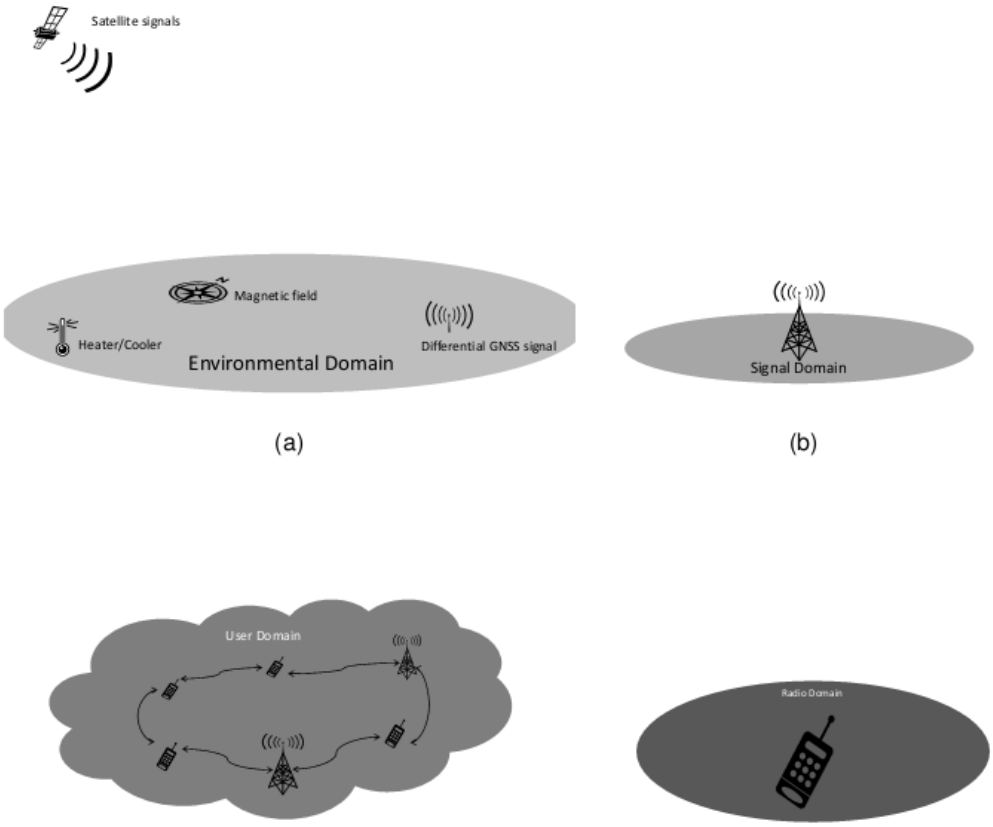


Figure 6-1 CRbDSA Domains: (a) Environmental Domain; (b) Signal Domain; (c) User Domain; and, (d) Radio Domain.

Environmental domain is the area in which significant knobs are perceived. An attack based on a specific knob may propagate through the environment in which mentioned knob is perceived. Environmental Domain is illustrated in Figure 6.1a.

Signal domain is the area representing the characteristics of the signal. An attack based on a specific spectrum band propagates through the area in which mentioned spectrum band signals are processed. Signal Domain is illustrated in Figure 6.1b.

User domain is the network among collaborative users and base stations. An attack based on cognitive radio network propagates through the collaborated users and base stations of this cognitive radio network User Domain is illustrated in Figure 6.1c.

Radio domain is determined by an operating radio. An attack aims at a specific radio propagates through this radio only. Radio Domain is illustrated in Figure 6.1d.

Although there are some exceptions, domains are mostly related to geographical area. Nominal surface of the Domains of CRbDSA is illustrated in Figure 6.2.

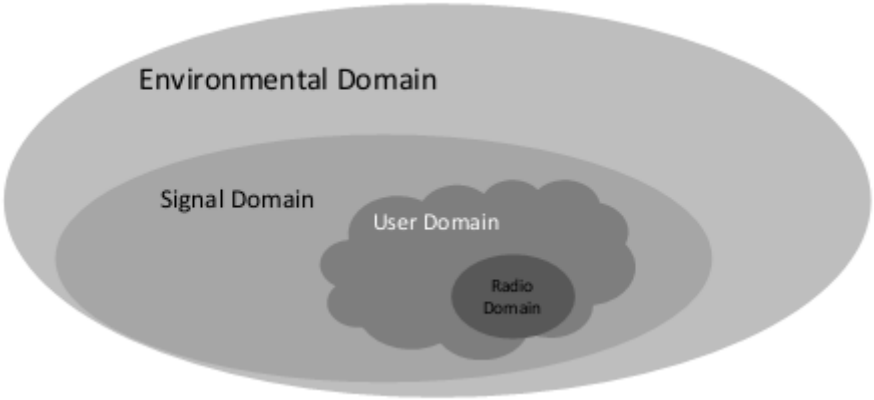


Figure 6-2 CRbDSA Domains

6.2 Complex attacks of CRbDSA

Although CRbDSA improves network and application performance and forms a collaborative network, specific resources and stages expose new classes of threats. CRbDSA enables attackers to combine more than one attack to generate a new class of threat, Complex Attacks. Before defining the Complex Attack, first the initiator of Complex Attacks, Triggers, should be defined. Following this, some of the known and potential Triggers are given in this section.

Triggers:
The attacks enable or provide a convenient medium to launch further attacks.

Complex Attack:
The total of the attacks initiated by Triggers. Besides the attacks realized in a convenient medium provided by Triggers are Complex attacks.

6.2.1 Known Triggers of CRbDSA

Although CRbDSA is aware of all the communication layers, physical and link layer have become the most significant issue. Almost every studied threat is based on physical and link layer of CRbDSA. The known and studied CR and DSA security threats examined in detail in [100], [101],[102],[103],and [104] are; Obstructing the whole spectrum by Jamming and Common Control Channel Attack, Intercepting the best scored channel by Maximal Interception Attack, Incapacitating the spectrum by Spectral Honey Pot, Usurping the vacant channels by exploiting Selfish Use, Masquerading the primary users by Mask Primary User attack and Hidden Node Problem, Falsification of the secondary users by Self-Propagating, Cognitive Radio Virus threats, SSDF, Beacon Falsification, and Primary User Emulation, Corruption of the cognition mechanism of radios by Policy Modification, Belief Manipulation and Objective Function Attack, Obstructing the sensing mechanism of radios by Obstruct Synchronization of QPs.

Jamming and Common Control Channel Attack
If an attacker sends strong radio signal to obstruct the whole spectrum, Jamming or Common Control Channel (CCC) attack occurs. Jamming consumes too much energy and jammers are easily detected. On the other hand, if the common control channel is known, the attacker may jam only this channel with relatively lower cost. Both attacks obstruct the whole spectrum. Common control channel must be known to exploit CCC attack. However, attacker may forge the MAC frames and guide users to a specific channel instead of CCC.

Jamming is influential at Physical layer on the other hand, due to requiring MAC frames; CCC attack is influential at Link layer. Both attacks obstruct the spectrum in the Act stage of the Cognition Cycle. Jamming propagates through the Signal domain whereas CCC attack propagates through the User domain.

Maximal Interception Attack
Users and base stations compute the scores of channel quality for messaging in the Observe stage of the Cognition Cycle. Attacker may capture these values from MAC frames and jam the best-scored channel. Maximal Interception Attack prevents collaborative users observing the environment.

Spectral Honey Pot
Aiming at incapacitating of the CRbDSA, an attacker conducts the radios by jamming until the target channel is located. This type of attack is defined as Spectral Honey Pot and propagating through the Signal domain. Spectral Honey Pot degrades the communication quality. Guiding a radio to target channel may be done by sweeping the band in whole or, in a smarter manner, transmitting the signal on the channels one by one until the radio locates on the target channel.

Selfish Use
If the malicious secondary users perceive the aforementioned link layer knobs and poses as a primary user by means of the gathered in-formation, they can gain access to the target channel. This type of attack is defined as Selfish Use and propagating through the Signal do-main. Selfish use prevents users from allocating channels.

Mask Primary User Attack and Hidden Node Problem
Despite being non-cognitive, primary users may also be exposed to CRbDSA attacks. An attacker may prevent users from realizing the existence of primary user by masking. This type of attack is defined as Mask Primary User attack. Collaborative users mostly do not experience with masking, since common sense is the main prevention method for this type of attack.
When a primary user signal is weaker than the remaining ones secondary users may not detect and interfere the primary user with higher signals. Hidden Node Problem is an unintentional interference threat but in a non-cooperative network, widespread. Mask Primary User and Hidden Node Problem propagate through the User and the Signal domain respectively. Both attacks prevent radios from perceiving the required knobs in the Observe stage.

Self-Propagating
If the behaviors are induced among each radio, an attacker may propagate a state and this state may spread among the other radios. If this state were not optimal, this would cause degradation of evaluation and generation of the alternatives in the Plan stage. This type of attack is defined as Self Propagating and propagating through the User domain. Trust mechanism and common sense algorithms may be employed to mitigate this type of attack.

Cognitive Radio Virus
If self-propagating behavior attack aims at a learning radio it can lead to the radio work sub optimally as a network virus. This type of attack, defined as Cognitive Radio Virus, can spread among non-cooperative radios. CR viruses propagate through the User domain.

SSDF
An attacker may capture, replay or falsify MAC frames and thus radios operate sub optimally or maliciously. Falsification of the radios leads evaluating mechanism operate sub optimally in the Orient stage. This type of attack is defined as Spectrum Sensing Data Falsification (SSDF) and propagating through the User domain.

Beacon Falsification and Policy Modification
In IEEE 802.22 [107] all inter-cell control messages (beacons) are vulnerable to be inferred, obstructed, falsified or replayed by unauthorized users. This type of attack is defined as Beacon Falsification Attack. Beacon falsification propagates through the User domain. Beacons provide policy update information for a period of time. A manipulated policy may lead to a malfunction in the Plan stage. Policy Modification is a conclusion of a beacon falsification attack and propagating through the User domain.

Primary User Emulation
Primary users have different features and secondary users are mostly aware of these physical layer knobs. In a selfish manner, an attacker may masquerade as a primary user to force the secondary user release the channel. This type of attack is defined as Primary User Emulation (PUE) and propagating through the Signal domain. PUE degrades evaluating and generating the alternatives in the Plan stage of the Cognition Cycle.

Belief Manipulation
During the learning period, if manipulated, sensory inputs are believed and a behavior is formed. This belief may lead to malfunctioning or malicious manner of radio. This type of attack is defined as Belief Manipulation Attack and propagating through the User domain.

Objective Function Attack
Learning algorithms usually have an objective function to be maximized or minimized under some constraint, weights and limits in the Orient stage. If an attack is well designed and aware of the objective function, it manipulates the inputs of the function and makes believe the radio that optimal is not good or malicious is not bad. This attack is defined as Objective Function Attack and propagating through the User domain.

Obstruct Synchronization of QPs
The attacks against primary and secondary users are also valid for the base stations. In addition to them, since they are the center authority of the CRN and charged as data fusion center, base stations are exposed to the falsification of the control data.
Quiet Period (QP) synchronizes the sensing period among the collaborative users who facilitate increasing of the sensing quality. Obstruct Synchronization of QPs is exploited to falsify the base stations by means of the modified beacons. This type of attack propagates through the User domain.
Aforementioned known Triggers are shown on Table 6.1.

Known Triggers of CRbDSA

Attack Name	Targeted resource	Targeted stage	Targeted layer	Effected Domain
Jamming	Spectrum	Act	Physical	Signal
CCC Attack	Control Channel	Act	Link	User
Maximal Interception	Most Sensed Channel	Observe	Link	User
Selfish Use	Data Channel	Decide	Physical	Signal
Mask Primary User	Primary User	Observe	Physical	Signal
Hidden Node Problem	Spectrum	Observe	Physical	Signal
Self Propagating	Users	Plan	Link	User
CR Virus	Users	Learn	Link	Radio
PUE	Users	Plan	Physical	Signal
Policy Modification	Secondary Users	Plan	Link	User
Belief Manipulation	Secondary Users	Learn	Link	User
Objective Function Attack	Radios	Orient	Link	Radio
Obstruc Sync. of QPs	Users	Observe	Link	User
SSDF	Users	Orient	Link	User
Beacon Falsification	Users	Decide	Link	User

Table 6-1 CRbDSA Triggers

6.2.2 Potential Triggers of CRbDSA

Physical and link layer have attracted more attention in security issues for CRbDSA. Commonly network and application layer security issues are disregarded. In fact, on the contrary of sensing and sharing stages of cognition mechanism, spectrum mobility and decision stages take advantages and disadvantages of all the layers of OSI reference model. Thus potential attacks of CRbDSA should be examined by considering network and application layer as well. As mentioned before CRNs are formed by including the network layer into cognition process. Cognitive routing is one of the network operations of CRN. Unlike traditional routing schemes, cognitive routing is aware of the spectrum allocation and this may cause new security threats to CRbDSA.
Inference from the communication layers by Traffic Analysis, Signal Analysis, Reverse Engineering,

Incapacitating the network by Black/Gray Holes, Usurping the network by Sybil Radios, Falsification of the environment by Environment Tampering, Obstruction of the evaluation mechanism of radios in the Orient stage of the Cognition Cycle by Multiplicity/Indeterminacy problem is potential simple attacks and threats for CRbDSA.

Signal Analysis
Despite detecting the masquerading users as a mitigation method to deception attacks, Signal Analysis may be exploited by the assailants to interfere information from the channels. Signal energy, spectrum usage and the other physical layer knobs may be featured as primary, secondary users and base stations. A cognitive skilled malicious radio can emulate as a primary user or save this information for a future attack. This type of attack propagates through the Signal domain.

Traffic Analysis
Traffic Analysis is performed for determining the activity of network, locations of base stations and users, protocol types of the networking or routing. Received and transmitted packet number and size, source and destination addresses, and type of the packets are decisive information about the activity of the network.
Type of the packets also determines the base station and the users' physical location information. This type of attack is very common among all the networks. Besides this, an attacker takes opportunity to gain more information in CRN than traditional networks by traffic analysis. Networking in CRbDSA is performed by means of cognitive process in CRNs. Analyzing the traffic may reveal information being exploited for a further attack as in signal analysis. Traffic analysis exploits the decisions of collaborative users thus propagates through the User domain.

Black/Gray Holes
Not only for degrading the quality of routing, a malicious user can attract or force the traffic to itself and then drop some or all the packets to it. Attackers may find much chance to force the traffic by interfered channels at cognitive routing. Dropping some of the packets is defined as Selective Packet forwarding or Gray Hole Attack, and dropping all the packets is defined as Black hole attack. Black and Gray Hole Attacks propagate through the User domain.

Sybil Radios
Cognitive Radio Networks nodes take advantage of the services provided by CRN namely optimal routing, fair channel usage, trusted communication. Radios must provide CRN with local information thus can make use of global services. Sybil radios use the services provided by the cognitive network but do not support CRN with efficient information. Sybil radios are the selfish users of network layer.

Environment Tampering
In contrast to misleading the sensors by sending false inputs, an attacker may affect the environment itself. A magnetization to deviate the north, or a heater to a temperature sensor may tamper the domain. In sensor networks, tampering the environment is a common threat although it is very difficult to implement. This type of attack propagates through the Environmental domain and falsifies the sensing mechanism of the radios in the Observe stage.

Reverse Engineering
The outputs of the learning engines are decisions about how to divide the spectrum into channels, which channels to use as control or data channel, routing policies and trusted and untrusted radios. To infer the framework of the cognitive network, a node can use this information. For an objective function attack radio must know the objective function, in the absence of this information an attacker may exploit a reverse engineering to understand which of the inputs affect the decisions. Especially classifier based learning algorithms are subjected to this type of attacks. For example, Channel quality scores may be captured by the Reverse Engineering. The inferences are based on link layer knobs and extracted in the Orient stage. Thus Reverse engineering propagates through the Radio domain.

Multiplicity/Indeterminacy Problem
Game theory may be employed to select the best channel, network or routing scheme among collaborative radios. Sequence of majority pairwise votes may lead to a cycle. For example, channel A may be better than B or channel B may be better than C or channel C may be better than A for some of the majorities. If cycles occur, multiplicity and in-determinacy problem reveals. Besides, Nash equilibrium for dynamic spectrum sharing is inefficient in some cases and this theory may lead to mentioned problem [108]. Multiplicity and Indeterminacy may be revealed unintentionally or by an attacker. This attack propagates through the User domain. This problem obstructs the evaluation mechanism of the radios in the Orient stage of the Cognition Cycle at link layer.
Aforementioned potential Triggers are shown on Table 6.2.

Attack Name	Targeted resource	Targeted stage	Targeted layer	Effected Domain
Traffic Analysis	CR Network	Learn	Network	User
Signal Analysis	Spectrum	Learn	Physical	Signal
Black/Gray Holes	CR Network	Act	Network	User
Sybil Radios	Users	Orient	Link	User
Environment Tampering	Radios	Observe	Physical	Environmental
Reverse engineering	Radios	Orient	Link	User
Multiplicity/Indeterminacy	Radios	Orient	Link	User

Table 6-2 Potential Triggers of CRbDSA

6.3 Classification of Complex Attacks

Complex Attacks are exploited by collaborative attackers or an attacker with various identities at the same time or produced in temporally and/or spatially sequences and stages. A Complex Attack may be exploited in two ways in CRbDSA.

- Dependent attacks in the stages of the Cognition Cycle,
- Dependent attacks to threaten any asset of CRbDSA by ascending through the communication layers.

The second method is defined as Cross-layer attack. However stages of the Cognition Cycle are not layer dependent in some cases, attacks to stages of the Cognition Cycle are considered as Cross-layer attack in literature. To make sense, attacks to stages of Cognition Cycle must be defined. Based on [105],

Cross-stage attack:
A Cross-stage attack is a collection of attack activities that are conducted coordinately in multiple stages of the Cognition Cycle in order to achieve specific attack goals.
There are numerous researches for Cross-layer attacks. In [105] a Cross-layer attack of CR, Lion attack, and mitigation method is proposed. Lion attack jams targeted users to reduce the throughput of TCP services (Round Trip Timeout, Round Trip Time) by forcing a frequency handoff.

Another Cross-layer attack, Jellyfish attack, is defined in [106]. Jellyfish attack is similar with Lion attack. Jellyfish attack starts with Spectral Honey Pot. Lion attack's starter simple attack is on Physical layer on the other hand Jellyfish attack's is on Link Layer. Figure 6.3 illustrates the Lion and Jellyfish attacks.

Due to Cross-layer design of CRbDSA, Cross-layer attack scenarios may be further developed. Cross-layer attacks may exploit more than two layers. For example, an attack to cognitive routing mechanism may be performed by three simple attacks. Start with an Environment Tampering to create a practical environment to exploit SSDF. Afterwards attacker may exploit a Black/gray hole attack by means of misled radios.

Attacks in any stage of the Cognition Cycle are layer independent. For example, contrary to common belief, the Observe stage is not only based on the Physical layer. Upper layer knobs are also perceived in the Observe stage. Two Cross-stage attack scenarios are developed to give an example. Link layer based attacks Self-propagating error and Cognitive Radio virus threats are exploited to form a Complex Attack.

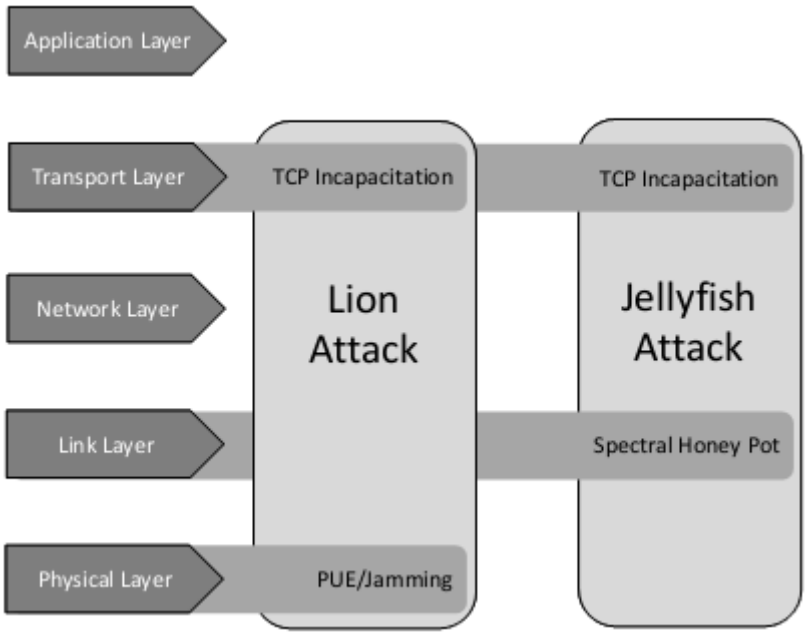


Figure 6-3 Sample Cross-layer Attacks

In the first scenario, attack starts with Self-propagating error in the Plan stage and in the second scenario attack starts with Sybil Radios threat in the Orient stage. First scenario is based on the study [100]. If Self-propagating error propagates through the collaborated users, attacker takes opportunity to damage the radio in the Learn stage as a virus infected computer.
The second scenario is based on tampering. An attacker makes targeted users believe an improper state is preferable by Sybil radios.

Figure 6.4 illustrates the mentioned Cross-stage attack scenarios.

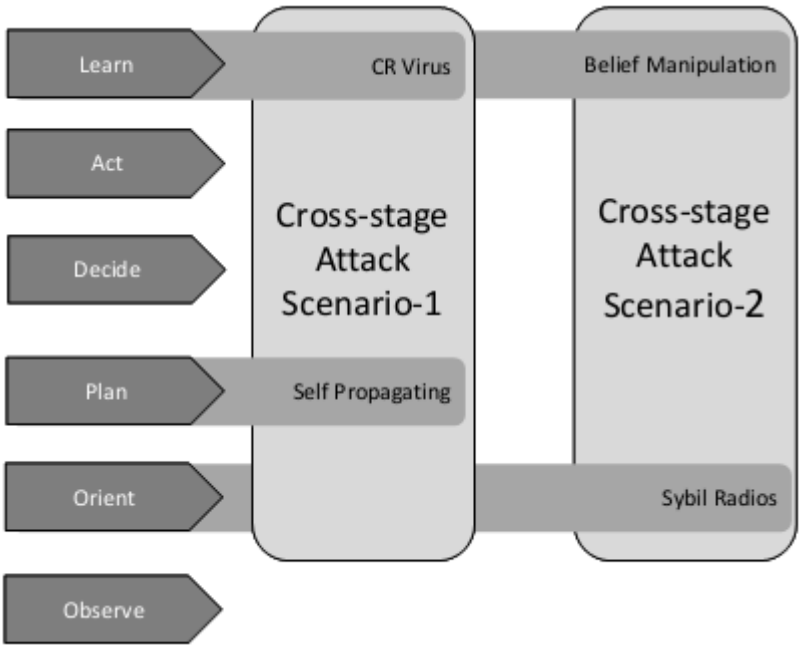


Figure 6-4 Sample Cross-stage Attacks

6.4 Defending against complex attacks

Defending against Complex Attacks includes detecting the Triggers by classifiers, making an attack more expensive to execute by prevention methods and reducing the damage from a successful attack by mitigation methods. To design security architecture for CRN and defense in depth against Complex Attacks of CRbDSA detection, prevention and mitigation methods are given in this section.

6.4.1 Detection Methods for Complex Attacks

IDSs' attack detection mechanisms are based on rules, signatures and anomaly detection methods. Rule based IDSs extract features and compare with pre-determined samples to expose attacks in detection process. Additional rules help IDSs detect Complex Attacks accurately.

A detected simple attack's probability of being a Complex Attack starter is increased if the attack is exploited in the early stages of the Cognition Cycle or lower communication layer. Besides, if the area of influence is getting bigger, the attack severity is increased as well. Each attack or threat may be classified by probability of being a Complex Attack starter. If an attack is based on a lower communication layer or an early stage of the Cognition Cycle, this attack has a higher probability of being a Complex Attack starter.

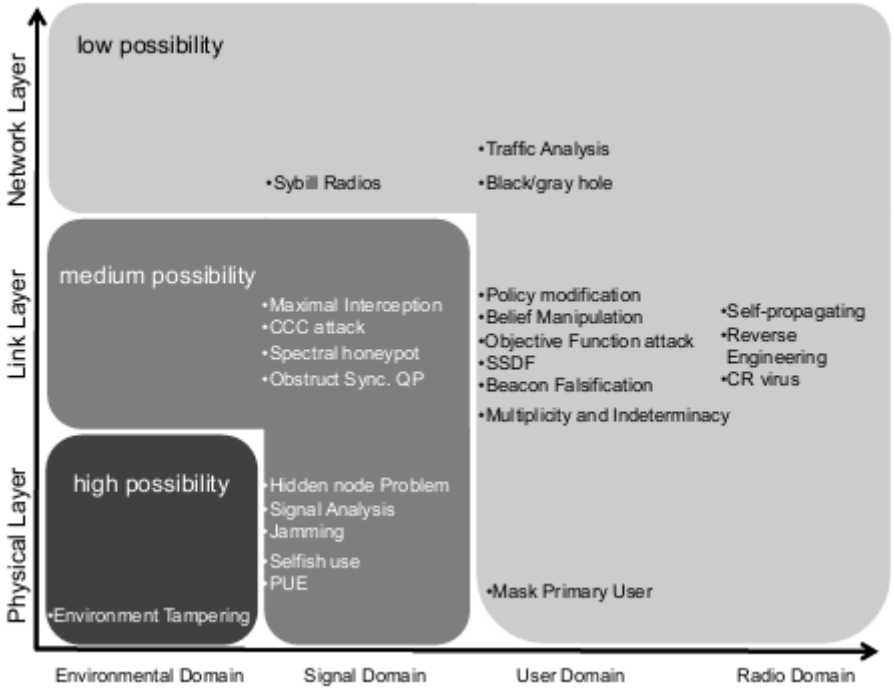


Figure 6-5 The Probabilities of Simple Attacks' Being a Starter of a Cross-Layer Attack

Examined simple attacks' Complex Attack probabilities are shown in Figure 6.5. Cognition Cycle stages are measured on(x) axis and domains are measured on(y) axis. An attack closer to origin has higher probability of being a Complex Attack starter.

Environment tampering has the highest probability of starter of a Cross-stage attack. As mentioned before exploiting a tampering attack to environment is not straightforward. However, location based CRbDSA may expose this attack easily. GNSS-based (Global Navigation Satellite System) positioning systems are exposed to replaying, relaying and jamming based attacks. Selective Availability (SA) still remains as a threat to GNSS. Besides, falsification of Assisted GNSS is an open issue.

In the same way as Cross-layer attacks, Cross-stage attacks may be classified. Sample Complex Attack starter probabilities are shown in Figure 6.6. Cognition Cycle stages are measured on(x) axis and domains are measured on (y) axis. An attack closer to origin has higher probability of being a Complex Attack starter.

Environmental tampering has still highest probability of being a Complex Attack starter based on Cross-stage. To form a structural classification low, medium, and high probabilities of being a complex-attack starter are formed and illustrated in Figure 6.5 and Figure 6.6.

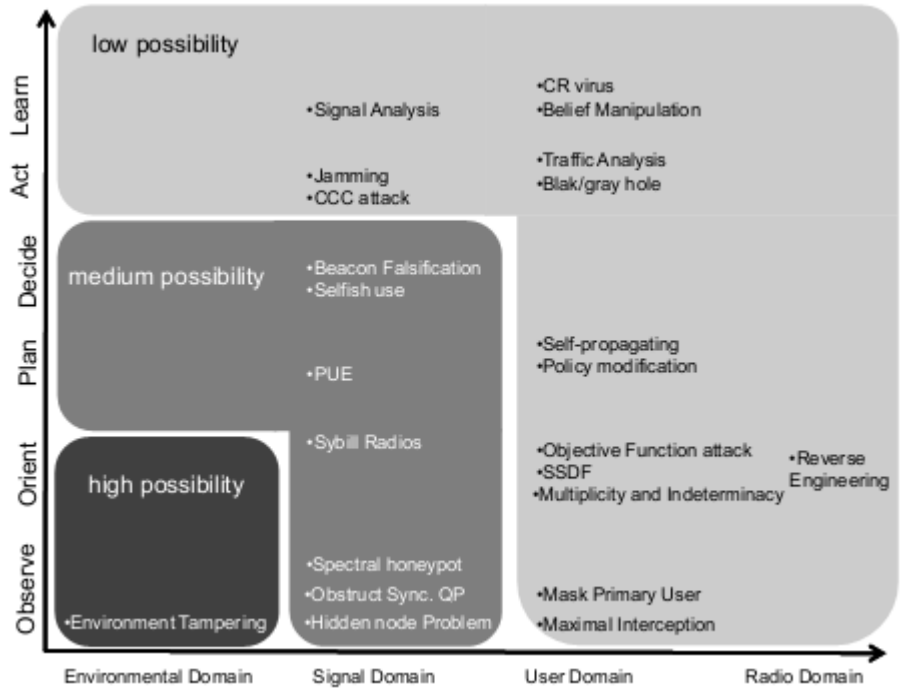


Figure 6-6 The Probabilities of Simple Attacks' Being a Starter of a Cross-Stage Attack

Classification rules to detect a simple attack's probability of being a Complex Attack starter are;

- If an attack occurs at the User or the Radio domain, this attack has low probability of being a Complex Attack starter,
- If an attack occurs at Network layer, this attack has low probability of being a Complex Attack starter,
- If an attack occurs in the Plan or the Decide stage and the Signal domain, this attack has medium probability of being a Complex Attack starter,
- If an attack occurs at Link layer and the Signal domain, this attack has medium probability of being a Complex Attack starter,
- If an attack occurs in the Observe or the Orient stage and the Environmental do-main, this attack has high probability of being a Complex Attack starter,
- If an attack occurs at Physical layer and the Environmental domain, this attack has high probability of being a Complex Attack starter.

6.4.2 Prevention Methods for Complex Attacks

Especially attacks starting at lower layer or early stages of the Cognition Cycle may reveal Complex Attacks. Early detection methods must be employed as prevention methods. Collaborated users are more robust to falsification-based attacks. Especially, due to environment tampering has highest probability of being a Complex Attack starter; common sense mechanism may prevent users from misleading. Jamming based Triggers reveal at Link layer and obstruct the sensing mechanism in the Observe stage. These Triggers propagate through the signal domain, thus Jamming based Triggers have the higher probability of being a starter. Mitigation methods for Jamming based Triggers are prevention methods for Complex Attacks. Direct sequence and Frequency hopping are Spread Spectrum techniques to pre-vent jamming. While direct sequence is used for continuous-time narrowband jamming, frequency hopping is used for pulse jamming. Coding techniques namely Rateless coding, Piecewise coding [109], distillation codes, erasure codes are used to mitigate for jamming effects.

6.4.3 Mitigation Methods for Complex Attacks

After a Cross-stage attack, the serious problem is manipulated radio in the Learning stage.
Belief expiration may be employed to mitigate for Cross-stage attacks. After a Cross-layer attack, mostly the cognitive routing is obstructed. As in Cross-stage attacks, routing table expiration may be considered in cognitive radio networks.

Next generation wireless networks are formed by contributing Cognitive Radio with Dynamic Spectrum Access technologies. Due to the specific framework of the CRbDSA, new class of attacks and threats emerge. To implement security evaluation in CRbDSA, in this study;

- Known Triggers, initiators of a Complex Attack, are re-examined in the context of the communication layers, stages of Cognition Cycle and new defined domains,
- Potential Triggers of CRbDSA are unveiled,
- Complex Attacks of CRbDSA are defined and classified,
- A kind of Complex Attack, Cross-stage attack, is defined.

Complex Attacks are formed by exploiting dependent simple attacks and each Complex Attack starts with a simple attack. Complex Attack starter probability can be assigned to each simple attack. A methodology is given to assign mentioned probabilities to each simple attack in this study. Intrusion detection systems must be aware of the simple attacks that have highest probability degrees.

Examined and defined simple attacks are classified into their probability of starter of a Complex Attacks. Finally, detection, prevention and mitigation methods of Complex Attacks are given. This study helps IDSs improve feature extraction and signature examination processes.

6.4.4 DDoS Countermeasures

There is currently no comprehensive method to protect against all known forms of DDoS attacks.
In addition, many derivative DDoS attacks are continually being developed by attackers to bypass each new countermeasure employed. Moreover, it must determine where the defense has to be deployed.

DDoS attacks have several features that hinder their successful detection and defense:

1. DDoS attacks generate a large volume how to overwhelm the target network.
2. It is difficult to distinguish attack packets from legitimate packets.
3. Most DDoS attacks use spoofed IP addresses.
4. The large number of attacking machines and the use of source IP address spoofing make the trace back difficult or impossible.
5. Although the router performs the ingress filtering, a lot of spoofing packets can pass it because some DDoS tools provide the several spoofing levels in order to pass the ingress-filtering router.
6. Distributed nature of the attacks calls for a distributed response, but cooperation among administrative domains is hard to achieve [60].

To build a suitable defense of a system against DDoS attacks one should know the counter-measures. In this section, a classification on DDoS countermeasures will be given and each class of countermeasures will be explained.

There is a plenty of papers that categorize DDoS defenses. Three categories of DDoS countermeasures introduced in [45]. Firstly, prevention of the setup of the DDoS attack network, including preventing secondary victims and detecting and neutralizing handlers. Secondly, dealing with a
DDoS attack while it is in progress, including detecting or preventing, mitigating or stopping, and detecting the attack. The post-attack category, which involves network forensic, discussed for the third. Other defense classification is based on activity level and location [44] and on submissive defense mechanism, active defense mechanism, action and defense deployment position [43].
Countermeasures described in this section will follow the classification proposed in [47]. Campagne et al gives a detailed countermeasure classification in three main categories: mitigation, deterrence and prevention.

6.4.4.1 Mitigation

Mitigating the effects of a DDoS attack does not necessarily require detection of the attack. A defense mechanism can be effective without knowing whether there is an ongoing attack or not.
Applying policies that isolate certain portions of traffic from others can limit the impact of malicious behavior without the need for an attack detection mechanism.
Network In network-based mitigation, we consider techniques involving the intermediate network as well as some mitigation methods assisted by client or server subnets.

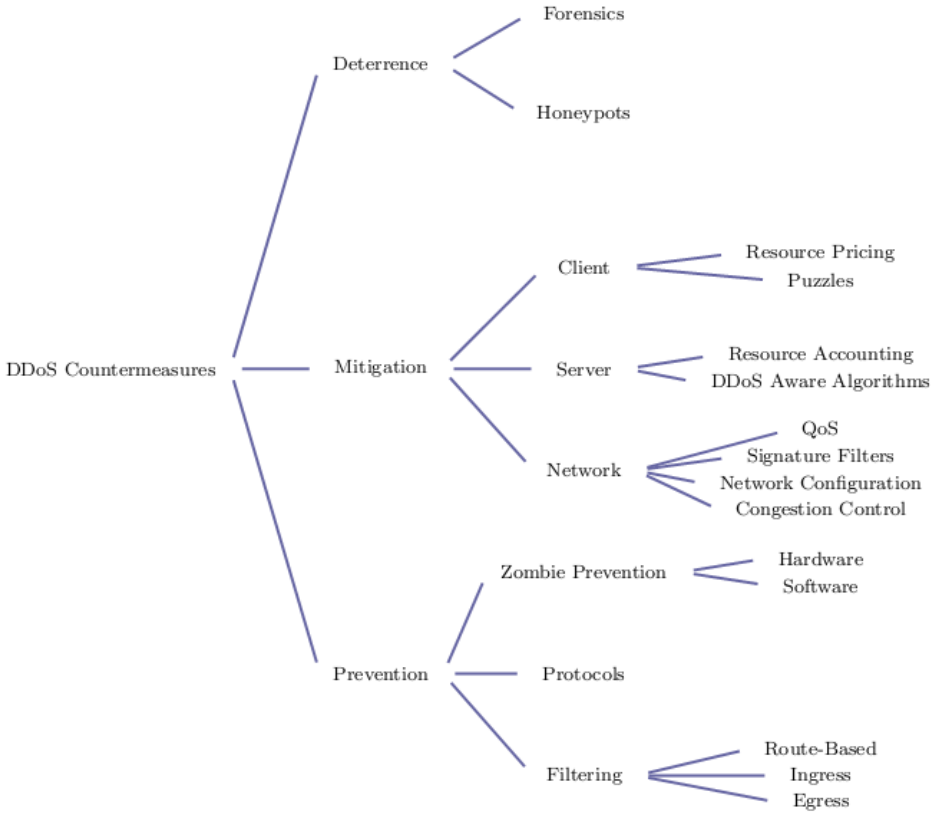


Figure 6-7 Taxonomy of DDoS countermeasures

Congestion Control
Such a congestion control can be done at different levels of abstraction, which we classify as link, flow or aggregate.

Link:
In a link-based congestion control scheme, a router maintains a queue for each incoming link. Forwarding of packets is done by sampling the head packet from each queue on a round-robin basis.

Flow:
One way to abstract traffic is to classify packets according to their network origins and destinations. The resulting classes of packets are called flows. Overloaded routers can be configured to throttle certain flows rather than a specific ingress link. This means routing policies do not affect a well-behaved flow sharing an ingress link with a misbehaved one.

Aggregate:
When controlling a large number of zombie machines, attackers have the possibility to produce attacking flows that individually appear well behaved. Such an attack is harder to detect but can have sufficient power to deplete available resources on the victim's network. Defense systems can thwart those attacks by classifying traffic with more precision than what is possible with flows. The resulting classes of packets are called aggregates.

An aggregate is defined as a group of packets sharing a common property [48]. Such properties include a packet's origin and destination as well as application or protocol type. For example, refined aggregates could consider only TCP SYN or ICMP ECHO packets. Isolating particular aggregates from the rest of the traffic allows more precise filtering that reduces the impact of DDoS attacks on unrelated traffic. This kind of protection can be used to mitigate bandwidth depletion attacks.

Network Configuration
Several schemes provide protection from DDoS attacks by modifying the physical or logical configuration of a network and its servers. Redundancy: Basic schemes introduce redundancy in order to increase server-processing capacity. The objective is to process all incoming traffic at any time so increases in a server's load do not affect any of its clients. Those techniques thus help a server withstand DDoS attacks as well as flash crowds. Flash crowds are sudden bursts in traffic due to a large number of content requests from legitimate users. Overlay:

Advanced methods require either adding an extra layer of networking components atop an existing infrastructure or extending the functionality of nodes already in place. In [49] a Secure Overlay Service (SOS) is provided by an overlay network of routers, which use a hash-based algorithm to route packets to a server. An outside host wishing to communicate with that server must first contact a Secure Overlay Access Point (SOAP), a designated router that lets a packet enter the overlay only after authenticating its source.

Roaming
Many of the known attack tools carry out attacks without performing DNS lookups: they send packets directly to a victim's IP address. Using this fact, a server under attack can modify its IP address. Doing so affects legitimate clients until they perform a new DNS lookup, but it protects the server from the malicious traffic still directed at the old IP address. If this DNS change can be connected to a detection unit such as IDS, the victim server can be protected automatically just after detection of DDoS attack. Therefore, service can be given new connecting legitimate users in very short time. For users that already connected to the server, can continue their usage when they refresh their connection to the victim server.

Signature Filters
Signature-based strategies maintain the traditional best effort routing model of the Internet in normal conditions. The defense system reacts only when an attack detection mechanism flags certain packets as malicious. Based on their reactive actions, signature-based mechanisms fall into two categories: local filtering and trace back mechanisms. Members of the latter category locate the source of the attack before taking action while local filtering methods immediately trigger countermeasures without knowledge of the attacker’s location.

Local Filtering:
Local filtering methods are distinguishable by the type of signature they use to classify traffic. When a node determines that packets with a certain signature are malicious, subsequent incoming packets with the same signature are discarded or rate-limited.

IP Trace back:
Trace back mechanisms form a subset of DDoS countermeasures that focuses on localizing the origin of a stream of attacking packets, regardless of whether their source addresses have been spoofed. Knowledge of the attacker’s location allows activation of filters closer to the source of the attack. This reduces the impact of collateral damage, where filtering nodes drop legitimate traffic identified as malicious. Such false positives are inevitable, but they are less likely to occur as filters affect a smaller part of the Internet.

Active trace back mechanisms recursively query upstream routers to obtain information about a certain malicious stream. With passive mechanisms, the intermediate network automatically sends path information to the victim. In both cases, the victim compiles the partial (or complete) path information to construct the sequence of routers used by a malicious packet

Active trace back mechanisms can be further divided into Memory less and History based approaches. The former category does not require the routers in the intermediate network to store information about forwarded packets whereas the latter category requires routers to store some information about each packet they forward.

(Active/Memory less: Controlled Flooding)
Burch and Cheswick [61] proposed one of the first techniques tracing IP packets back to their source. In this method, the node under attack first requests generation of a brief burst of UDP packets on each link connected to the closest router. The overload due to the temporary increase in traffic causes the router to drop packets. Downstream, the victim monitors the effect of flooding each link. A decrease in attacking traffic concurrent to the flood of a certain link indicates an attacker located further upstream on that particular link.
This link is thus added to the list of links forming the path to the attacker. The search for the attacker then goes on by recursively applying the method hop by hop. During that process, links for which flooding has no effect are pruned out of the exploration space.

Passive trace back methods can be divided into two subcategories: message-based methods on the one hand and header marking methods on the other hand. In message-based methods, nodes on the path of a packet generate extra packets that contain tracing information. With header marking techniques, routers include path information within the header of an IP packet.

ICMP trace back, an early method proposed in [62] uses such a message-based scheme. With a low probability (typically 1/20,000), routers participating in ICMP trace back send ICMP messages to the destination addresses of the packets they forward.

An ICMP message contains the IP address of the generating router. It also contains either the IP address of the router from which the generator received the packet of interest, or the IP address of the router to which the generator forwarded the packet of interest, or both. A control message thus consists of one or two edges in the graph representing the path of a packet. In order to form an attack graph, a host targeted by an attack compiles all the edges corresponding to packets it considers malicious. The resulting graph is a tree with the victim as a root and attackers as leaves.

QoS
A possible way to cope with attacks targeting depletion of network bandwidth is to introduce a service differentiation mechanism that reserves a share of the bandwidth to certain categories of traffic. Such a mechanism creates classes of packets that are each treated differently by the network (higher priority classes are forwarded first). Ideally, only legitimate packets get preferential treatment and all attacking traffic belongs to the lowest class of service. However, in real-life situations, such precise categorization is hard to achieve.

Server
Several strategies investigate how to defend a server from DDoS attacks by modifying the server itself. These techniques deal primarily with software. There exist simple ways for an operating system to mitigate the effects of a DDoS attack. For example, an OS can periodically scan the TCP connection queue and drop half-open connections. By doing so, the OS prevents a TCP SYN attack from hogging memory resources. Lazy Receiver Processing (LRP) can also help an operating system in the case of a flooding attack by avoiding certain computations on packets that end up being dropped due to overload.

Some OS-level resource accounting schemes like Escort [51] are more elaborate than simple DDoS-aware algorithms. They enforce policies, which control allocation of time-multiplexed resources such as CPU time or network bandwidth.

Client Puzzles and Resource Pricing schemes are client-centric classes of countermeasures that take advantage of the limited computational and monetary resources of client hosts in order to force them to regulate their traffic. With puzzles, every client requesting access to services must commit a certain amount of resources determined by the network or server. Resource Pricing strategies establish different market-like schemes in which resources are available for purchase by the clients.

6.4.4.2 DDoS Prevention

Reactive measures protecting critical services during an ongoing attack e.g., the majority of measures are undoubtedly important, but ultimately, proactive measures need to be carried out to prevent the occurrence of an attack in the first place. Such preventive measures should either eliminate exploitable flaws on a network or work towards complicating the task of a potential attacker.

Filtering
When internetworking protocols such as TCP and IP were initially designed, functionality was of greater concern than security. As a result, malicious parties can generate invalid information or send harmful commands without being detected by those protocols. Filtering packets with such contents is a first step towards reducing an eventual enemy's perniciousness.

Protocols
How to design protocols that do not offer opportunities for DoS attacks is still an unsolved research problem. However, there are some desirable properties known to prevent specific types of attacks. For example, the goal of some attacks is to deplete a server's resources by establishing a large number of bogus TCP connections. That way, the TCP buffers are saturated with connection status and incoming connection requests must be ignored. A remedy to this problem involves designing stateless protocols, which shift the burden of state holding from the server to the clients. SYN cookies partially reach this goal by remaining stateless in the first steps of a TCP connection establishment.
Zombie Prevention: An important step towards solving the DDoS problem consists in preventing the attacker from constituting an army of zombie computers in the first place. To attain this goal, it is necessary to remediate the weaknesses attackers exploit to gain control of hosts connected to the public Internet. One of the most important weaknesses, buffer overflow [55], can be mitigated using either software or hardware mechanisms.

6.4.4.3 Deterrence

Even though some techniques allow the tracing of an attack back to the attacking hosts, a victim will rarely be able to identify and prosecute the attacker controlling those zombies. Any method altering this climate of impunity might deter some malicious parties from waging DDoS attacks.

Honey pots
Honey pots are computer systems placed on a network for the sole purpose of being abused by unsuspecting attackers. Since a honey pot does not offer any useful service, nearly all activity detected on it is malicious. It is thus simple to use such a computer as an intrusion detection system.
In regular operating systems, root kits can be used to cover the attacker's traces. In advanced honey pot systems, the OS is encapsulated in a logging framework, so all attacker activity is recorded, regardless of attempts to alter the audit trail. The possibility to track the attacker's every move can be a deterrent since the attacker may not want to expose his tactics. Honey pots can only be useful to avoid attacks from script kiddies. Script kiddies are inexperienced hackers that use scripts available to public for hacking. Experienced attackers can be aware of honey pots and do not attack to these machines.

Forensics
With the use of custom-made sniffers and scripts, skilled programmers can manually trace the activity of malicious software back to the IRC channel used by the master attacker for controlling the bot network [61]. Such forensic activity could ultimately lead to the discovery of the attacker's identity.

6.5 PHY layer security in MIMO-OFDM

The well-known network model of the Open Systems Interconnect (OSI) reference model proposes a composition of seven layers; physical, data link, network, transport, presentation, session and application layers, distributes network's functionalities to distinct layers that are assumed to independently from other layers. Using the OSI reference model can provide a formal definition and practical terms that affects information security on a layer-by-layer basis. Security can be seen as an aggregation of protection mechanisms of different layers.

A conceptual visualization of layered security solutions is shown in Figure 6.8. One should pass through all the layers in order to acquire private data. In such cases, Physical layer (PHY) security becomes inevitably important, as it forms the first step of the security system. In this section, information theoretical approaches, which are also included in the initial PHY security studies, will be introduced. This will be followed by a discussion of physical vulnerabilities of wireless systems. Common physical layer attacks, namely eavesdropping; traffic analysis, jamming, message modification, information disclosure, masquerade, ID theft, man-in-the-middle and denial of service attacks will be introduced. State of the art PHY security methods will be classified into two; the code based methods and the signaling based methods, along with the corresponding attack types that they aim to prevent. Code based methods include error correction coding (ECC) and spread spectrum coding (SSC) details of which will be given.

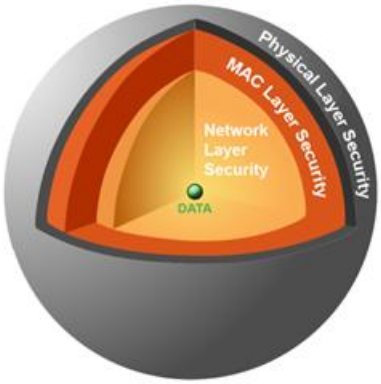


Figure 6-8 Layers of data

Signaling based methods include beamforming and artificial noise methods, which are proven to be effective countermeasures for privacy attacks, are therefore very important. *Beamforming* is a multi-antenna technique that enables the transmitter to focus signals spatially. *Artificial noise* (AN) is a recent concept that is utilized in PHY layer security methods, consisting of transmitting noise signals generated by the transmitter to non-legitimate users to degrade their signal reception quality. There are two main AN approaches in the literature, namely isotropic AN and smart AN. Isotropic AN approach is based on broadcasting the generated noise without spatial selectivity except legitimate user's direction, whereas the smart AN approach is based on sending AN only to the locations and/or frequency bands where eavesdropper exists. This section will be concluded with the open issues about implementations of PHY security countermeasures.

6.5.1 System Model and Security Performance Metrics

Security is an important issue in wireless networks due to the open nature of wireless medium. Several studies have been conducted to improve security systems for wireless networks. As a consequence, there exist many solutions offered in different layers. Here, a generic system model will be given along with a background on PHY layer security techniques and the related work.

First, we will introduce the general signal and channel model of a wireless system in order to define performance metrics. The most basic wireless system with eavesdropping can be designed to have one transmitter node A, one legitimate receiver node B and an eavesdropper node E. Assume that all the nodes are equipped with single transmitting or receiving antenna for simplicity. Data bits are coded and modulated before transmission regarding to the selected modulation and coding scheme. Let $s(k)$ be the data signal that A wants to send to B and let $x(k)$ represent the signal to be transmitted. For now, we assume A transmits only the data signal, without weighting or noise addition, so $x(k) = s(k)$ is transmitted signal from A. The received signals of A and E are given as $r_B(k)$ and $r_E(t)$ and they are defined as,

$$r_B(k) = x(k)\, h_{AB}(k) + n_B(k) \quad (1)$$

$$r_E(k) = x(k)\, h_{AE}(k) + n_E(k) \quad (2)$$

where $n_B(k)$ and $n_E(k)$ are the additive zero-mean Gaussian noise (AWGN) components and $h_{AB}(k)$ and $h_{AE}(k)$are the channel coefficients of the channels between nodes A and B and nodes A and E, respectively. Estimates of these channel coefficients are referred to as channel state information, and modeled as a standard real AWGN channel. There are different channel models that are frequently being used in the literature such as Rayleigh, Rician or Nakagami-m fading channel models, as well as off-the-shelf solutions such as Stanford University Interim (SUI) or 3GPP WIM2 models. Channel effects as path loss, shadowing, multipath and Doppler shift are included in these models. Channel noise powers of legitimate user's channel and eavesdropper's channel are defined as σ_B^2 and σ_E^2 , respectively.

A very useful measure of the channel quality is signal to interference and noise ratio (SINR). This ratio gives how strong the received data signal power compared to non-data signal power caused by channel noise and interference.

$$SINR_B(k) = P_{signal}\, |h_{AB}(k)|^2 / \sigma_B^2 \quad (3)$$

$$SINR_E(k) = P_{signal}\, |h_{AE}(k)|^2 / \sigma_E^2 \quad (4)$$

where the average transmit signal power is defined as P_{signal}. Note that SINR is equal to signal to noise ratio (SNR) when interference is interference power is zero, however it is SINR definition is important for the security system models with interference, given in the following sections.

In the assumption of $\sigma_E > \sigma_B$, the *secrecy capacity*, highest transmission rate at which the eavesdropper is unable to decode any information, is defined by,

$$C_{secrecy}(k) = C_B(k) - C_E(k) = \frac{1}{2}\log_2(1 + SINR_B(k)) - \frac{1}{2}\log_2(1 + SINR_E(k)). \quad (5)$$

Notice that in order for secrecy capacity to be non-zero, $SINR_B(k)$should be higher than $SINR_E(k)$, meaning $N_E > N_B$ should be satisfied. As in [110] a complex AWGN channel is equal to two parallel real-valued AWGN channels, as a result the secrecy capacity complex AWGN can be defined by,

$$C_{secrecy}(k) = \begin{cases} \log_2(1 + SINR_B(k)) - \log_2(1 + SINR_E(k)), & \sigma_E > \sigma_B, \\ 0, & otherwise. \end{cases} \quad (6)$$

The upper bound of perfectly secret transmission rate from the source node to legitimate destination node is defined as the *secrecy rate*. Secrecy capacity is also the achievable maximum secrecy rate. The probability of outage in secrecy capacity is another important definition of information theoretic analysis of PHY layer security. *Outage secrecy capacity* (OSC) probability is defined as the probability that the instantaneous secrecy capacity being less than a target secrecy rate as

$$P_{OSC}(R_S) = P(C_{secrecy} < R_S) = P(C_{secrecy} < R_S \mid SINR_B > SINR_E)\, P(SINR_B > SINR_E) + P(C_{secrecy} < R_S \mid SINRB \le SINR_E)\, P(SINR_B \le SINR_E). \quad (7)$$

After the transmitted signals arrive at the receiving antenna, the received signals are demodulated and decoded to bits, where it is possible to calculate the bit error rate (BER) of the system, which is one of the primary performance measures for digital communication systems. Usually a minimum BER requirement is defined for a successful communication, depending on the desired application. If BER of a system is below a minimum required level, a communication ling cannot be properly established. As a result, it can be seen that satisfying a non-sufficient BER on unauthorized nodes can actually provide security. Hence, BER can also be used to define the quality of service (QoS) and the PHY layer security level of a system. It is obvious that SINR value of the channel is directly related to system BER, however, this relation varies according to the preferred modulation and coding schemes. For every system, higher SINRs point to lower BER values. In order to avoid the effect of modulation and coding techniques, SINR can be used for the definition of QoS and PHY security levels. In the next section, a more detailed model of a PHY layer security system will be given. For more information about channel models and wireless communication systems basics, readers are referred to additional reading section.

6.5.2 State of the Art

Most of the attacks in PHY layer can be categorized as eavesdropping-based attacks. Eavesdropping attacks are unauthorized compromise of the data traffic between the between legitimate nodes. Traffic analysis attacks are an example of eavesdropping-based attack type, where the content of the data is not compromised but the transmitter and receiver nodes are detected. These types of attacks are typically approached via cryptographic algorithms that are implemented at higher network layers. However, the new approach is to prevent the attack in PHY layer, by using physical characteristics of the wireless channel for secure communication.

In the pioneering work of Wyner [111], the wiretap channel is introduced as seen in Figure 6.9(a) and it is shown that when an eavesdropper's channel is a degraded version of the main communication channel, perfect secrecy, as defined by Shannon [112] can be achieved.

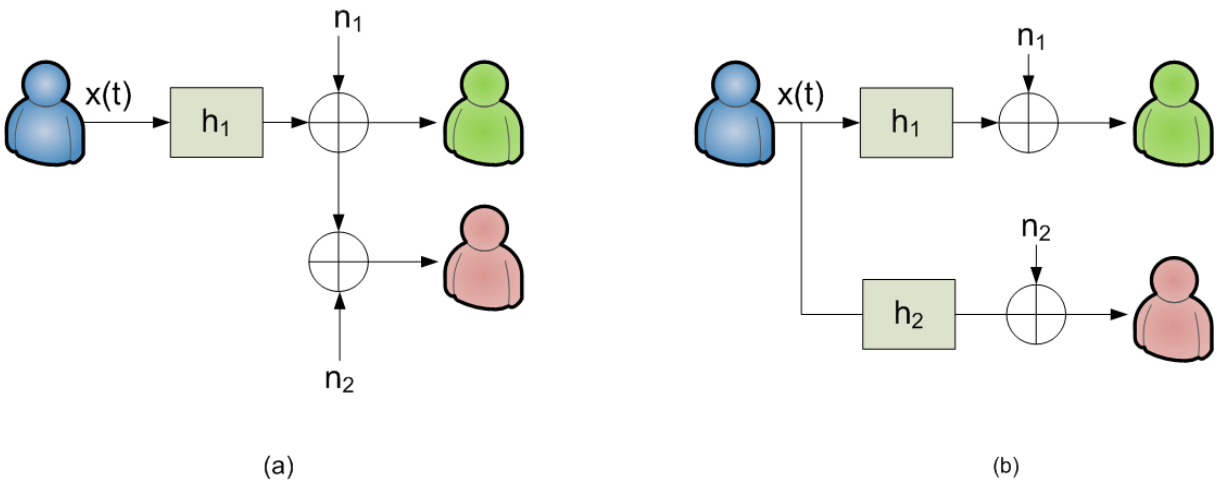


Figure 6-9 Channel model of a system with eavesdroppers, (a) Wiretap channel model of Wyner, (b) independent channel model.

In the study [113], authors considered a general independent channel condition as seen in Figure 6.9(b), by eliminating the degraded eavesdropper channel assumption and studied the transmission. The results of this study generalized the results of [111], however it was stated that the channel of eavesdroppers should be still noisier than that of legitimate receiver's. These early studies showed that positive secrecy capacity could be achieved for a wireless communication system in the presence of eavesdroppers with noisier channels. The impact of these works remains limited until the arrival of enabling technologies such as smart antennas, increased computational capabilities of electronic devices and multi-input multi-output (MIMO) systems. The main reason was the requirement of the legitimate transmitter and receiver to have some advantage over the attacker in terms of channel quality, which cannot be guaranteed in a practical system. Moreover, almost at the same time, Diffie and Hellman published the basic principles of public-key cryptography [114], which was adopted nearly all security systems.

Later, in 2000s multi-antenna systems have enabled a very useful technique, referred to as beamforming. The first beamforming study for security discussed preventing jamming attacks by deploying beamforming [115]. In this study, the use of directional antennas is shown to give higher performance than omni-directional antennas. A single input multi output (SIMO) system model is considered in [116]. In this study, beamforming transmission can provide maximization of secrecy rate in Gaussian channels, in multi-antenna systems. The secrecy capacity analysis in MIMO systems are conducted in [117] and [118]. These studies assume that full channel state information (CSI) is provided, which may not be very practical to assume for real-world scenarios. However, it is shown that full CSI at transmitter case is an upper bound for secrecy rate [119], meaning the secrecy rate will decrease in partial CSI systems. Following the lead of these studies, many theoretical and practical studies have been conducted to analyze the secrecy rates of different systems. The majority of these studies used information theoretic approaches and the *secret channel capacity* or *secrecy capacity bounds* have become a very common performance metric of PHY layer security systems.

The ergodic secrecy capacity in fading channels is examined with and without CSI of eavesdropper. In [120], a fading broadcast channel is deployed and a scheme to find the optimal power allocation that minimizes the outage secrecy probability is proposed. Furthermore, various physical-layer techniques were proposed to achieve secure communication, even if the receiver's channel is worse than the eavesdropper's channel. One of these techniques is the use of interference or artificial noise to confuse the eavesdropper. With two base stations connected by a high-capacity backbone, one base station can simultaneously transmit an interfering signal to secure the uplink communication for the other base station [121]. In the scenario where the transmitter has a helping interferer or a relay node, the secrecy level can also be increased by having the interferer to send random noise signals independently at an appropriate rate. This scheme is called as cooperative jamming. When multiple cooperative nodes are available to help the transmitter, the optimal weights of the signal transmitted from cooperative nodes, which maximize an achievable secrecy rate, were derived for both decode-and-forward protocols in [122] and amplify-and-forward protocols in [123]. The use of interference for secrecy is also extended to multiple access and broadcast channels with user cooperation [124].

With the use of multi-antenna systems, it is possible to simultaneously transmit both the information bearing signal and the artificial noise to achieve secrecy in a fading environment as shown in [125] and [126]. In these studies, the artificial noise is transmitted isotropically and CSI of the eavesdropper is not required. However, it should be noted that the legitimate communication also gets affected by the transmitted noise. As a result, the transmit power allocation on data signals and AN are crucial in these systems. A suboptimal power allocation strategy was considered in [127], which aims to meet a target signal-to-interference-and-noise ratio at the intended receiver to satisfy a quality of service requirement. In the study [128], a scenario with multiple antenna eavesdroppers is deployed. The authors in have shown that when selection combining (SC) is used one multiple antenna eavesdropper causes the same effect as of multiple single antenna eavesdroppers. Moreover, it was shown that as the number of the eavesdropper's antennas increases, the secrecy outage probability also rises. Many recent studies exist that make use of convex optimization techniques to find the optimal AN and beamforming weights that satisfy SINR with minimum power. In [129] space selective artificial noise is proposed and two minimization problems are built. The first minimization problem is to minimize total power while satisfying SINR constraints on eavesdropper on legitimate receiver and eavesdropper, while second problem is to maximize SINR of legitimate receiver for a given maximum SINR for

eavesdropper and limiting the total power. The proposed space selective artificial noise scenario shown to be more power-effective compared to isotropic AN and no-AN scenarios, however the space selectivity requires the CSI of eavesdropper to be known by the transmitter. In [110], this idea is further developed by changing the position of the AN source from the transmitter to the receiver. Authors state that self-interference cancellation techniques are used at receiver in order to eliminate the effects of the simultaneous artificial noise transmission on the receiver antennas. The results are shown to be effective compared to AN at transmitter systems under perfect self-interference cancellation, which is very hard to achieve.

The second direction of using PHY layer attributes in encryption systems, which is referred as PHY based key generation in [130]. This idea is based on channel reciprocity, which is the term for equality of transmitter to receiver and receiver to transmitter channel responses, in other words, uplink and downlink channels are equal. This specialty allows two communicating nodes to share a unique random data, as channel state information between two nodes cannot be gathered by any other nodes if it is not broadcasted. This random data can provide a common randomness source to system nodes. Eavesdroppers may only be able to detect the channel between transmitter and themselves, but there is no way they can estimate the channel between two legitimate nodes. As a result, the keys are provably secure with information theoretic guarantee and PHY-based encryption key generation has become a very promising area of research. Especially in mobile systems, common random source is very dynamic due the fast changing nature of the wireless channel. However, the slow changing environments may have limited source for key generation. Inevitably, generated key rate depends on the frequency of the changes in the channel.

6.5.3 Physical Layer Security in Wireless Networks

PHY Layer security has become popular with the arising of wireless technologies. Wireless medium is potentially unsafe as the communications signals are broadcasted into air and anyone in the antenna range can access the transmitted signals. In traditional wired technologies, the possibility of the transmit signals are gathered by an untrusted third party may not be a main trouble, as physical security is deployed by hiding cables in walls and cable endpoints locked up in server rooms or cabinets. If one, use a special device to “line in” to the link, a physical attempt need to be done. Hence, even though it is not impossible, there is a certain difficulty for gathering the signals from a cable unless you have the endpoint. As a result, traditional systems usually accept that cable is secure, meaning that it is assumed that no one can access the data unless they access the endpoints. However, this assumption cannot be made for wireless systems, as wireless signals, unlike cables, are available to every node in a range equipped with a proper receiving antenna.

6.5.3.1 Major Security Requirements in Wireless Networks

The main requirements of a wireless security system are authentication, secrecy and data integrity along with robustness to physical attacks like jamming or natural effects like channel noise or interference. We detail these requirements below.

Secrecy

Secrecy (data secrecy), in a communications system refers to the state that the information is obtainable solely by the legitimate receiver. This is a challenge that should be properly addressed especially for wireless communication systems. In wired communication systems, data secrecy is accepted to be guaranteed between two nodes, which means a cable is assumed to be secure and security is considered to be maintained on the network nodes on the way from sender to receiver. In another words, it is often accepted that if sender and receiver is directly connected by cable, then the data cannot be obtained by anyone else so secrecy is maintained on PHY layer. However, as mentioned, wireless medium has an open nature that makes it very hard to maintain secrecy. Any receiver in the coverage of sender antenna can capture the communication signals without being noticed. In wireless systems, non-legitimate receivers can execute such an attack, eliminating secrecy of data. In such case, maintaining physical security becomes very important.

Major attack types against data secrecy are eavesdropping and traffic analysis. Eavesdropping is the act of secretly listening to the private conversation of others without their consent, which projects to gathering of wireless communication data by non-legitimate users. Eavesdropping attacks are typically very easy to perform and very challenging to detect due to their passive nature. Such attacks can be performed with a proper receiving antenna and a system for decryption of the encrypted data, if encryption is made. Traffic analysis is a similar version of eavesdropping, in which the non-legitimate user cannot intercept the communication data but gathers the traffic information, like sender and receiver identities, data rates, data type, data protocols. Usually traffic analysis attack is performed where the encryption key cannot be gathered. Major countermeasures to eavesdropping and traffic analysis attacks are encryption, beamforming and artificial noise. These security techniques will be introduced and the detailed review will be given in the following section.

Authentication

User authentication is a means of identifying the user and verifying that the user is allowed to access some restricted service. Proper authentication mechanisms can be considered as the base of the security expedients, because of their importance. If a non-legitimate user can be authenticated by the system, every restricted service and information can be easily accessed. Moreover, the risk of the information and/or the system to be altered is highly considerable, for example, a wireless remote system can be severely damaged causing destructive incidents, depending on the application.

Authentication is usually executed by using an authentication key mechanism. Authentication keys can be obtained from ID-based systems, hardware tokens, channel coefficients, pre-shared passwords and location information.

ID-based cryptography, which is introduced in[131], is rapidly emerging in the recent years. Now, identity based encryption (IBE) algorithm in [132] and combined public key (CPK) algorithm in [133] are the two popular identity based cryptography systems. In IBE, user identity is its public key, and it cancelled the chain of certification authority (CA), but it still requires online databases. CPK-based authentication systems do not need online database or trusted third party, which result in improved processing and efficiency ability. Nan Xiang-hao first proposed CPK in 1999 that overcomes disadvantages of IBE and awarded a national patent in 2006 [134]. CPK can judge user identity by its key. The theoretical foundation of CPK is based on the elliptic curve discrete logarithm problem (ECDLP). In identification applications, CPK-based authentication system is expected to get increased attention in the near future.

Major authentication attack types can be outlined as brute-force attacks, eavesdropping attacks, man in the middle attacks, authentication cloning and ID theft attacks. Man in the middle (MIM) attacks are a form of active eavesdropping in which the attacker makes independent connections with the target nodes and messages between them, making them believe that they are talking directly to each other over a secure connection. In fact, during a MIM attack, the entire conversation is controlled by the attacker. Beyond the secrecy violation, it is clear that MIM attacks can be very dangerous to systems as the attacker is authenticated and it is able to enter the system or change the communication data in a harmful manner. In an authentication cloning attack, an unauthorized user pretends to be a legitimate user by deceiving the authentication system. An authentication cloning attack can be implemented in many ways, including capturing the authentication sequences that are based on PHY layer attributes. For example, an intruder can imitate its location or channel information as the legitimate user and are authenticated to access resources. Identity theft occurs when an attacker captures network traffic and identifies the MAC address of a device with network privileges. Most wireless systems allow some kind of ID filtering to allow only authorized device with specific IDs to gain access and utilize the network. However, devices exist that have network “sniffing” capabilities, which mean they are able to capture the transmitted data in a network. If these devices can imitate the authorized devices' ID as their own, they can easily be authenticated. Identity information can also be gathered by executing brute force attack, which means trying all the possible ID key options [135].

Data Integrity Awareness

In its broadest meaning, data integrity refers to the trustworthiness of information over its entire life cycle. It is the representational faithfulness of information to the true state of the object that the information represents. Representational faithfulness has four essential attributes: completeness, currency/timeliness, accuracy/correctness and validity/authorization. Integrity is a critical requirement in wireless systems, because of the potential vulnerabilities originated from PHY layer. Major data integrity attacks are message modification and jamming attacks. Attacker can send forged control, management or data frames over wireless to mislead the recipient or facilitate another type of attack. Message modification is the general class of attack types that based on additions or deletions to actual data by malicious users. Jamming attacks are based on transmitting signals to depress or degrading the communication service performance. Jamming attacks usually aim to block legitimate communication, but can result in partial corruptions in data as well. Authentication based attacks can also lead to data integrity issues, as altering data is a possible action once the attacker gets authenticated. In order to detect data integrity issues, integrity checks like checksum checks are performed. If data is intentionally altered with a message modification attack, it is not likely to be detected. However, integrity checks are still an efficient way to deal with PHY layer errors during transmission. After integrity issue is detected, correction algorithms can be used as error correction coding (ECC) systems to correct some erroneous parts of data, a request for retransmission can be made.

Robustness

High degree of robustness against jamming and/or natural performance degrading effects such as noise or interference resulting from other wireless transmitters is one of the major design goals of wireless systems. The major attack type that a system should be robust against is denial of service (DoS) attack type. A DoS attack aims exhausting the available resources to system's legitimate users. DoS attacks may focus on any resource of a system in order to degrade or cancel service functionality. Jamming is widely used to execute DoS attacks at the physical layer. DoS attacks are sometimes executed by a number of distributed nodes to increase effectiveness and to reduce the risk of being detected, hence being prevented. These types of DoS attacks are named as distributed denial of service (DDoS) attacks and are accepted to be one of the most challenging security issues in communication systems.

Countermeasures of DoS or DDoS attacks are not clear as these attacks can be executed in various ways. Anomaly detection systems are used to determine if an attack is being held for a resource. If the system decides that there is a DoS attack, it usually prevents the attacker by blocking its resource usage. This countermeasure is harder in DDoS attack scenarios as the attacker divides the attack into number of different nodes, therefore it is very hard to tell if any node is an attacker or not. Another way to enhance the robustness of a system is to use resource diversification techniques, which means having back up resources to use if one resource is under attack. For example, if a jammer is detected, the system switches to a different center frequency to avoid the quality degrading effects. Usually systems are designed to have back up communication lines in different networks to avoid connection losses. Robustness can also be achieved in the device side, wireless systems can be designed to have back up devices that can be switched to, if the master device is under a physical attack.

6.5.4 Existing Solutions and Recommendations

In this section, we will present solutions to deal with aforementioned system vulnerabilities and attack types. We categorize the solutions to achieve maximum secrecy as code based methods, signaling based methods and PHY-based encryption methods.

6.5.4.1 Code Based Methods

The main objective of code-based approaches is to improve resilience against jamming and eavesdropping. These approaches include the use of error correction coding and spread spectrum coding.

Error Correction Coding

In a conventional cryptographic method, a single error in the received cipher text will cause a large number of errors in the decrypted plain text after channel decoding. In order to address this problem, a combination turbo coding and advanced encryption standard (AES) cryptosystem was proposed in [136]. This scheme uses the encrypted turbo codes to set up a secure communication session based on the pseudo-random number generation algorithms. Depending on channel conditions, this method can be adopted to choose the number of redundant bits required to protect the information in order to achieve higher efficiency. The main advantages of secure turbo codes include higher-speed encryption and decryption with higher security, smaller encoder/decoder size, and greater efficiency.

Spread Spectrum Coding

Spread spectrum is a signaling technique in which a signal is spread by a pseudo-noise (PN) sequence over a wide frequency band with frequency bandwidth much wider than that contained in the frequency bandwidth of the original information. Spread spectrum is an effective solution to achieve physical layer security. Direct sequence spread-spectrum (DSSS) has been widely used to spread the transmitted data over multiple frequencies [137]. Frequency-hopping spread-spectrum (FHSS) continuously changes the central frequency of a conventional carrier several times per bit duration (i.e., in a fast hopping system) in accordance with a randomly selected channel so that it is extremely difficult to illegally monitor the spread spectrum signals. The main difference between conventional cryptographic systems and spread-spectrum systems lies in their key sizes. Readers are referred to [138] for more information about this method.

6.5.4.2 Signaling Based Methods

Data protection can also be facilitated using signaling design approaches. The usual schemes in these approaches involve beamforming and the injection of artificial noise. It can be seen that beamforming and AN methods can be used to improve secrecy and can be used as a countermeasure to secrecy or authentication targeted attacks. These methods are detailed below.

Beamforming

Beamforming is a multi-antenna technique that enables the transmitter to focus signals spatially. Recently, beamforming techniques have received great interests and it can achieve performance and capacity enhancement without the need for additional power or spectrum. Beamforming is a type of radio frequency (RF) management in which an access point uses multiple antennas to send out the same signal. By sending out multiple signals and analyzing feedback from receivers, the wireless LAN infrastructure can adjust signals it sends out and can determine the best path the signal should take in order to reach a receiver node. In a sense, beamforming shapes the RF beam as it traverses the physical space. Early implementation of beamforming is switched beam technique, which refers to system that selects the optimum of pre-determined antenna patterns. A more dynamic way to spatially form the patterns is to use adaptive beamforming systems. These two systems are shown in Figure 6.10(a) and Figure 6.10(b). In switched beam scenario, the nearest pattern is chosen to the location of legitimate receiver. However, it should be noticed that the beam pattern is not perfectly focused on the intended target. Another drawback of switched beam is that side lobes cannot be controlled for eavesdropper locations, which is shown in Figure 6.10(b). These drawbacks are resolved in adaptive beamforming systems, in which the weights of antenna arrays can be changed dynamically. As seen in Figure 6.10(a), beam pattern can be designed as perfectly focused on the legitimate user and eavesdropper can be avoided.

In security studies, adaptive beamforming scenarios are deployed. However, it should be noted that switched beam systems also provide a secrecy improvement, compared to traditional isotropic antenna systems.

In a beamforming scenario, transmitter device has multi-antenna transmitter and applies beamforming to the information signals transmitted to authorized receiver. An unauthorized receiver cannot gather information signal as the signal is focused spatially on the authorized receiver. Therefore, beamforming is a major concept in PHY layer security, due to the secrecy it provides.

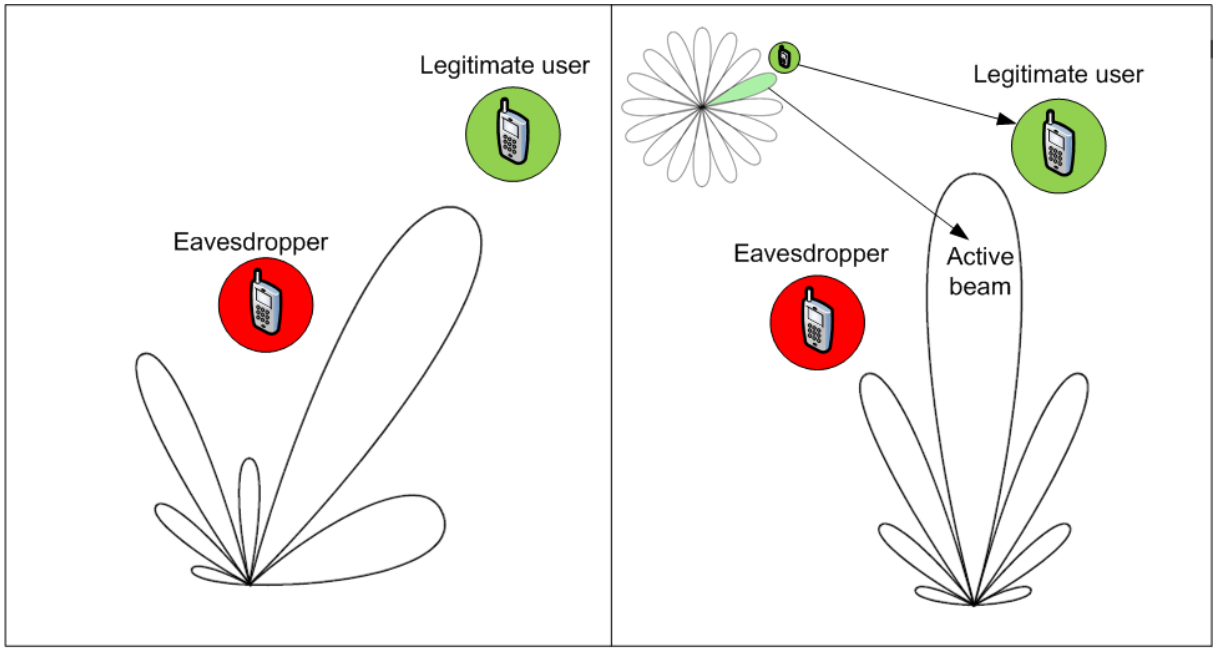


Figure 6-10 Beamforming, (a) adaptive beamforming, (b) switched beam system

Solely beamforming type of optimization scenario is a basic beamforming weight adjustment case, in which signal beam is targeted to legitimate user and not sent to anywhere else (If this problem is formed to minimize total transmit power, it can be expressed as a convex optimization problem and can be solved easily with convex optimization methods.

A system model of a beamforming system is very similar to the basic model that we have introduced earlier. Consider the system model with one legitimate transmitter (node A), one legitimate receiver (node B) and one eavesdropper (node E). This time the nodes are equipped with multi-antenna transmitter or receivers, with the quantity of N_A, N_B and N_E antennas, respectively. Hence, the transmitted signal $x(k)$ now will be multiplied with N_A sized weight coefficient matrix, $\mathbf{w}$. The received signal vectors of legitimate receiver and eavesdropper will be,

$$\mathbf{r}_B(k) = \mathbf{w}(k)\, s(k)\, \mathbf{H}_{AB}(k) + \mathbf{n}_B(k) \quad (8)$$

$$\boldsymbol{r}_E(k) = \mathbf{w}(k)\, s(k)\, \mathbf{H}_{AE}(k) + \mathbf{n}_E(k), \quad (9)$$

where $\mathbf{H}_{AB}$ is the channel matrix of MIMO channel between A and B having dimensions of $N_A \times N_B$ and $\mathbf{H}_{AE}$ is the channel matrix of MIMO channel between A and E having dimensions of $N_A\, x\, N_E$. The channels are assumed to be quasi-static in the time range k, which means the change in channel CSI through time is infinitely slow through k, so $\mathbf{H}_{AB}(k) = \mathbf{H}_{AB}$ and $\mathbf{H}_{AE}(k) = \mathbf{H}_{AE}$. With this assumption, we can also calculate $\mathbf{w}(k)$ for every different channel conditions i.e. when the location of receiver changes, so we have $\mathbf{w}(k) = \mathbf{w}$ for the time range k. The terms $\mathbf{n}_B(k)$ and $\mathbf{n}_E(k)$ are AWGN components, having length of N_B and N_E respectively. $\mathbf{r}_B(k)$ and $\mathbf{r}_E(k)$ have also length of N_B and N_E, defining one signal value for each receiver antenna. The received signals can be combined by many combining techniques. Assume maximal ratio combining (MRC) is deployed, which is the optimal combiner for AWGN channels as it maximizes the SINR by combining the received signals according to their channel noise level. SINR of an MRC receiver output is equivalent to the sum of all receive antenna SINRs.

Now we can define the SINRs as,

$$SINR_B = \sum_{i=1}^{N_B} P_{signal} \frac{\mathbf{w}^{\mathrm{H}} \mathbf{R}_{H_{AB,i}} \mathbf{w}}{\sigma_B^2} \quad , \quad SINR_E = \sum_{j=1}^{N_E} P_{signal} \frac{\mathbf{w}^{\mathrm{H}} \mathbf{R}_{H_{AE,j}} \mathbf{w}}{\sigma_E^2} \tag{10}$$

where $\mathbf{R}_{H_{AB,i}} = \mathbf{H}_{\mathrm{AB,i}}\ \mathbf{H}_{\mathrm{AB,i}}^{\mathrm{H}}$ and $\mathbf{R}_{H_{AE,j}} = \mathbf{H}_{\mathrm{AE,j}}\ \mathbf{H}_{\mathrm{AE,j}}^{\mathrm{H}}$ are defined as the instantaneous CSIs of B and E available to A. The notation of $\mathbf{H}_{\mathrm{AB,i}}$ refers to the N_A-length channel vector of A to i^{th} antenna of B. The SINRs are defined for each antenna of the receiver antenna array.

After this, a simple minimization problem can be solved by,

$$\min_{w} \quad ||\mathbf{w}||^2 \tag{11.a}$$

$$s.t. \quad SINR_B \geq \gamma_b \tag{11.b}$$

$$SINR_E \leq \gamma_e \tag{11.c}$$

where γ_b and γ_e are minimum and maximum SINR constraints that we require on B and E, respectively. By solving this optimization problem, we can obtain the weight vector $\mathbf{w}$, which satisfies the required SINR on legitimate receiver, a required secrecy level for eavesdropper with minimum power.

As the beam width is inversely proportional to gain in a directional antenna, directional transmission can improve spatial reuse and enlarge the geographical coverage. Under jamming attacks, use of directional antennas and beamforming methods may also become advantageous. Ander a jamming attack the node would still be able to receive data from the directions not covered by the jamming signals. Therefore, the employment of directional antennas can improve robustness also, by avoiding physical jamming attempts, and enhancing data availability.

Artificial Noise

Artificial noise is a novel concept that is utilized in PHY layer security methods, consists on sending generated noise by transmitter to non-legitimate users to degrade their signal reception quality. This method showed that perfect secrecy could be achieved when the intruder's channel is noisier than the receiver's channel. In this method, artificial noise is generated using multiple antennas or the coordination of helping nodes, and is injected into the null-subspace of the intended receiver's channel. AN is utilized to impair the intruder's channel, but it does not affect the intended receiver's channel since the noise is generated in the null-subspace of the legitimate receiver's channel. It was also shown in [139] that relying on AN, secret communications can be achieved even if the intruder enjoys a much better channel condition than the intended receiver.

Combined Beamforming and Artificial Noise

Beamforming and AN are proven to be effective PHY layer security methods. AN and beamforming can be used to maintain a defined secrecy level for eavesdroppers along with a defined service quality level for legitimate users. These methods can be divided to 2 categories as isotropic AN designs and smart AN designs.

In AN aided transmit beamforming scenarios, the transmit vector can be defined as

$$\mathbf{x}(k) = \mathbf{w}(k)s(k) + \mathbf{z}(k), \tag{12}$$

where $s(k)$ is the complex data stream that legitimate transmitter wants to send to legitimate receiver, $\mathbf{w}$ is the transmit beamforming weight vector of $s(t)$ and $\mathbf{z}(t)$ is the generated artificial noise vector with the length of N_A It is assumed that $\mathbf{z}(k) \sim CN(0, \boldsymbol{\Sigma})$, which means $\mathbf{z}(k)$is a random vector following a complex Gaussian distribution with mean 0 and covariance $\boldsymbol{\Sigma} > 0$.

Isotropic AN designs

In isotropic AN design, the transmitter generates a determined amount of artificial noise to interfere eavesdroppers. In isotropic AN case, the locations and channel conditions of eavesdroppers are not known and generated noise is transmitted everywhere but the legitimate user (null space of the legitimate receiver's channel), in an isotropic manner to improve secrecy.
In isotropic AN design, the transmitter generates a determined amount of artificial noise to interfere eavesdroppers. In this method AN covariance can be chosen as $\mathbf{\Sigma} = \beta \mathbf{P}_h^{\perp}$ where $P_h^{\perp} = I_{N_t} - \mathbf{R}_{H_{AB}}/||\mathbf{H}_{AB}||^2$ is the orthogonal complement projector of $\mathbf{H}_{AB}$, and $\beta > 0$ is a scale factor determining the power invested on AN. It is shown in [129] that if we choose ρ and β as

$$\rho = \frac{\sigma_n^2 \gamma_b}{||h||^4} \tag{13}$$

$$\beta = max\left\{0, \max_{m=1,\dots,M} \frac{\rho h^H R_{g,m} h - \sigma_{v,m}^2}{Tr(P_h^{\perp} R_{g,m})}\right\}, \tag{14}$$

then the optimum signal transmit weight vector and AN covariance can be obtained for a $N_B = N_E = 1$ single input multi output (SIMO) system, by

$$\boldsymbol{w} = \sqrt{\rho} h \quad , \quad \Sigma = \beta P_h^{\perp}. \tag{15}$$

Smart AN designs

Smart AN is a term that we use to categorize selective AN types. The generated noise can be formed selective in frequency, space and time, depending on the scenario. For example, it has been shown that pilot jammers are more effective then plain jammers on OFDM systems [140]. It is a reliable countermeasure for eavesdropping like attacks, by performing jamming attack to malicious users. Please note that this scenario uses more optimized transmit by eliminating the eavesdropper-free areas in the selectivity domain. However, it is not very practical to assume that location of eavesdroppers' is known in any domain, as eavesdropping is a passive attack type. However, the idea is that AN can be designed as focused on important parts of the transmission so that same secrecy can be provided with less power consumption. Optimization problems with smart AN cases can be designed in theory and as a result, these designs give better results than previous AN designs. The power optimization problem of space-selective AN can be briefly given as,

$$\min_{\mathbf{W},\mathbf{\Sigma}} \quad \|\mathbf{w}\|^2 + Tr(\mathbf{\Sigma}) \tag{16.a}$$

$$SINR(\mathbf{w}, \mathbf{\Sigma}) \geq \gamma_B \tag{16.b}$$

$$SINR(\mathbf{w}, \mathbf{\Sigma}) \leq \gamma_E \tag{16.c}$$

$$\Sigma \geq 0 \tag{16.d}$$

The last thing that should be discussed is the solution of the optimization problems mentioned above. The problems (11) and (16) are unfortunately non-convex due to the $\mathbf{w}^{\mathrm{H}}\mathbf{R}_H\mathbf{w}$ terms in the constraints on SINR of B. A common approach to deal with non-convex problems is to obtain a convex problem approximate the original solution by relaxing the non-convex constraints with semi-definite relaxation (SDR) techniques. A crucial first step in deriving an SDR of these problems is to observe that

$$\mathbf{w}^{\mathrm{H}}\mathbf{R}_H\mathbf{w} = \mathrm{Tr}\left(\mathbf{w}^{\mathrm{H}}\mathbf{R}_H\mathbf{w}\right) = \mathrm{Tr}\left(\mathbf{R}_H\mathbf{w}\,\mathbf{w}^{\mathrm{H}}\right) \tag{17}$$

Notice that defining new variable $\mathbf{W} = \mathbf{w}\mathbf{w}^{\mathrm{H}}$ is equivalent to $\mathbf{W}$ being a rank one symmetric PSD matrix, which adds the constraints $\mathbf{W} \geq 0$, rank $(\mathbf{W}) = 1$. However, the problem is still considered as very hard to solve due to the non-convex rank($\mathbf{W}$) = 1 constraint. Applying SDR approach to make the problem convex, we relax it by neglecting this constraint and we achieve the convex SDR problem as,

$$\min_{W,\Sigma} \quad Tr(W) + Tr(\Sigma) \tag{18.a}$$

$$s.t. \quad \frac{1}{\gamma_b} Tr(W R_{H_{AB}}) - Tr(R_{H_{AB}} \Sigma) \geq \sigma_B^2 \tag{18.b}$$

$$\frac{1}{\gamma_e} Tr(W R_{H_{AE}}) - Tr(R_{R_{H_{AE}}} \Sigma) \leq \sigma_E^2 \tag{18.c}$$

$$\Sigma \geq 0, W \geq 0. \tag{18.d}$$

As we omitted the non-convex rank-1 constraint, the problem in (18) is not actually equal to the original problem in (16). However, it can be verified that the solution of the SDR problem yields to the exact solution of the problem by using the rank reduction result of Lemma 3.1 in [141]. Thus, we can compute the FDB secrecy optimization problem with a formulation that can be solved, in an efficient and numerically reliable fashion.

PHY aided Encryption Key Extraction

Key exchange problem is one of the main challenges of encryption systems. The same encryption keys must be obtained in transmitter receiver pair for correctly decrypting the encrypted messages transmitted. However, if the key is transmitted through the insecure communication channel, the key can be gathered by non-legitimate users and encryption process becomes worthless as everyone can decrypt the transmitted signals especially in the shared wireless communication channels. In [114] Diffie and Hellman proposed a key exchange and key sharing method to solve the problem of key exchange. The purpose of this method is to allow only the authorized receiver transmitter obtain the encryption key, even if other nodes are able to hear the transmitted signals. Variations of this method have been proposed since then.

In almost all key exchange algorithms, length of public keys are chosen long, leading to increase in computational complexity and shortening of battery life on wireless devices [142]. In recent years, the idea of merging the physical layer attributes has emerged and a new class of key sharing technique is discovered [143]. Many studies have been done on different physical layer key exchange methods, which seem to have various advantages over the classical algorithms [144],[145]. These security methods can be classified in both data link layer and physical layer, merging advantages of encryption and physical layer privacy and enhancing existing encryption methods with physical layer components.

6.5.5 Future Research Directions

PHY layer security is a rather novel concept, implying that there are many opportunities for researchers. Future research directions include new and more effective smart beamforming and AN techniques. Seeking of PHY layer countermeasures against specific attack types of wireless network is also a solid need of research. Practical implementation is also an open area of research for transferring PHY layer security techniques from theory into real world systems.

6.6 IP based complex attack prevention

Attacks on IP networks can be detected by an Intrusion Detection System (IDS) and can be blocked by an Intrusion Prevention System (IPS). As described previously an IDS can use many methods, including statistical ones, to detect an intrusion. There are only few constraints on the time and the resources needed to perform all analyses because the only aim is to detect an intrusion, even if it takes a lot of time. On the other hand, an IPS is designed to prevent an intrusion. If malicious IP traffic is detected, it must immediately be dropped. In the case of an IPS, analyses are time and resources dependent because the network packets are temporarily blocked until being explicitly marked as non-malicious packets. Too complex analyses will introduce high traffic latency. Therefore being able to face complex attacks with very quick analyses is a huge challenge.

6.6.1 State-of-the-Art in Intrusion Prevention Systems

There are two common approaches to detect and block attacks over the network: *the protocol validation* and *the signature approach*.

6.6.1.1 Protocol validation

The key idea behind protocol validation is to ensure that what we see over the wire follows the corresponding standards and specifications. Concerning network protocols, standards are often described in RFC.

The first step is to be able to recognize dynamically the network protocol used, even if it does not use its default port. Then, each characteristic of the protocol is verified and validated. For example with the HTTP protocol, it is possible to ensure that all the headers have the right syntax and the right separator, the length of the URL is not anomalously big, no forbidden non-ASCII character is found in a header, and so on. It is possible to verify things that are strictly required to have a working protocol as well as things that are advised in a regular usage of the protocol.

This approach is proactive. Indeed, exploiting vulnerability leads often to a strange behavior in the protocol that can be detected. As a result, if the protocol is validated to block anything strange, an unknown future attack is likely to be prevented.

6.6.1.2 Signatures

The signature approach can be proactive, but in many cases it is reactive. It can be used when we have a known malicious content and we want to teach the IPS how to recognize and block it. There are two main ways to implement signatures: to use hash function or to use regular expression.

A hash function can be used for example on a binary file and act as a footprint. Then, this footprint can be compared to a virus database in order to know if the analyzed file is a known virus or not.

On the other hand, regular expressions can be used to find more precisely some wanted characteristics of a malware. Typically, a computer infected by a malware will send HTTP packets to a Command Control server in order to retrieve orders. The syntax of those requests often follows a pattern that can be detected accurately with regular expressions. This is a very flexible and convenient way to exclude a malicious content without blocking legitimate traffic.

Moreover, signatures can be proactive in some cases. For example generic signatures can be created to block XSS and SQL injections attacks with regular expressions in some specific contexts. As a result, it will cover quite all future XSS vulnerabilities that will be found in websites and products.

6.6.1.3 Netasq's approach

Netasq takes advantage of both protocol validation and signature approaches. Once a protocol is dynamically identified, the traffic will be inspected in depth in order to ensure that everything is valid. A footprint will be taken on some specific contents and passed to a virus database to identify and block many known threats. For that task we rely on our partnership with Kaspersky Antivirus and on the free antivirus ClamAV. On top of that, the protocol will be dissected in many contexts and regular expression signatures will be applied to each context to be able to block known malicious content.

6.6.2 Difficulty to face complex attacks

Nowadays, vulnerabilities are often complex and exploits can be highly sophisticated. A protocol validation coupled with a signature approach can be not enough.

Here, we will study an example of a recent vulnerability that impact Internet Explorer. This vulnerability is known under the identifiers CVE-2012-1889 and MS12-043. This is a *classic* ActiveX vulnerability where a COM object can be loaded via Internet Explorer and a vulnerable function of this object can be called with a specific payload to trigger the exploit.

Such vulnerability is quite easy to exploit while it is really complex to block without false positives.

Why is it so difficult? It is because there are plenty of ways to trigger the vulnerability.

CVE-2012-1889 is vulnerability in Microsoft Windows XML Core Services. To exploit it, one of the vulnerable objects must but load in Internet Explorer. In the examples below we will use MSXML2.DOMDocument (CLSID: F6D90F11-9C73-11D3-B32E-00C04F990BB4). Then a specific function of this object must be called. Here we will use a fake one in order to avoid the spread of working exploits. Let's call this function “run_exploit”.

6.6.2.1 HTML + JS

The regular way to load an ActiveX object is to use an <object> HTML tag with a given classid. Then this object is retrieved and used in JavaScript code.

```
<html>
<head><title>HTML and Javascript</title></head>
<body>
<object id='foo' classid='CLSID:{F6D90F11-9C73-11D3-B32E-00C04F990BB4}' ></object>
<script>
var obj = document.getElementById('foo');
obj.run_exploit(42);
</script>
</body>
</html>
```

Here we can notice that the protocol validation approach will not be useful because nothing is strange at the HTTP level.

The signature approach will may be hard too because the attack is split in two parts. The first one is in HTML language and the second one in JavaScript.

6.6.2.2 Pure JS

It is also possible to do the same thing without using HTML, in pure JavaScript.

```
<script>
var foo = document.createElement('object');
foo.setAttribute('classid','clsid:f6d90f11-9c73-11d3-b32e-00c04f990bb4');
foo.run_exploit(42);
</script>
```

A regular expression signature approach is therefore quite difficult because of the very different possible ways to do exactly the same thing. On top of that, using the classid of an object is not the only method available. Indeed we can also use the object name:

```
<script>
var foo = new ActiveXObject('MSXML2.DOMDocument');
foo.run_exploit(42);
</script>
```

6.6.2.3 Obfuscated JS

The previous examples were very simple. But in the *real life*, JavaScript code is often highly obfuscated by the attackers to evade IPS protections. Here, there are no limits to what is possible to do. Multiple layers of obfuscations, with usage of many intermediate variables, characters encoding and obscure functionalities of JavaScript language.

For example, here is a not-so-obfuscated example of the previous variant:

```
<script>
var c = 'fromCha';
var bar = 'MS' + String[c+'rCode'](88,77,76,0x32, 0x2e) + unescape('%44%4f%4d%44%6f%63%75%6d%65%6e%74') ;
var foo = new ActiveXObject(bar);
foo.run_exploit(40+2);
</script>
```

So the Javascript code should be normalized before any pattern matching verification.

6.6.2.4 Vbscript

Code obfuscation is not the only evasion technique. Indeed the use of Javascript is not mandatory. With Internet Explorer, we can do the same thing using VBscript language. Here is an equivalent of the first example:

```
<html>
<object id='foo' classid='CLSID:{F6D90F11-9C73-11D3-B32E-00C04F990BB4}' ></object>
<script language='VBScript'>
sub FOOSUB()
BAR=42
foo.run_exploit BAR
</script>
</html>
```

6.6.2.5 Multi connections (external JS)

Last but not least, it is possible to execute the same attack but split in multiple HTTP connections by calling an external JS script.

```
<html>

<object id='foo' classid='CLSID:{F6D90F11-9C73-11D3-B32E-00C04F990BB4}' ></object>

<script src='http://www.example.com/exploit.js'></script>

</html>
```

Here we will have two HTTP connections. Each one is not harmful if taken separately. But the combination of those two connections will launch the exploit.

To conclude, we have seen that vulnerability trivial to exploit can be in the meantime very hard to detect and block reliably. Indeed, all previous evasion methods can be combined to forge an undetectable attack.

6.6.2.6 Multi connections and multi protocols

Some attacks or products are using multiple protocols and connections and can be blocked only if each protocol and connection are analyzed and correlated. Let's take here the example of Skype.
Skype is a well-known VoIP and chat product that is able to work well even if the client is behind a Firewall.

Indeed, Skype can takes advantage of encryption, can use open ports like 80 or 443, uses DNS resolution but has got a huge hard-coded list of super nodes. If one communication channel is filtered or block, Skype will quite always find another way to reach the Internet.

An UDP or TCP forged by Skype cannot be detected if it is taken alone. It is encryption and there are not enough clues to recognize Skype. However, if we correlate events like a Skype DNS resolution followed by encrypted UDP/TCP packets sent to the resolved IP address, then it is possible to guess that these packets are related to Skype.

Skype can consume many bandwidths in an internal network, but it is not malicious. However there are other products that have the same kind of ability to reach the Internet and that is a real threat. For example products like Ultrasurf that can be used as a proxy to bypass a filter policy or Botnets that are using these mechanisms to contact Command & Control servers.

6.6.3 Future Research Directions

6.6.3.1 Ability to correlate multiple contexts

Netasq already have the possibility to split packets (in that case an HTTP packets containing HTML and JavaScript) into pertinent contexts in order to apply signatures on each of them:

- Inspection of HTML tags and attributes to be able to found an instance of a given ActiveX object
- Inspection and normalization of JavaScript code to be able to found known patterns even if the JavaScript code is obfuscated

The idea is to be able to correlate those contexts in order to build high-level, flexible and powerful signature capabilities. Thus many attack parts can be found separately in different places and different languages, and then correlated together to understand that an attack attempt is launched and to be able to block it.

6.6.3.2 Ability to correlate multiple connections

The next step is to be able to correlate multiple contexts of multiple connections using multiple protocols, e.g. in order to cover cases such as external JavaScript or to detect advanced attacks or specific products that cannot be blocked without that.

6.7 State-of-the-Art on Intrusion Response

In contrast to detecting and reporting, intrusion response aims to perform explicit actions that eliminate an on-going intrusion attempt or limit its impacts. The response process thus adds the ability to counter an attack by enforcing retaliation measures. Intrusion response has been longly superseded by intrusion detection. Only the last decade has experienced growing trends towards enhancing the response process, as discussed in the Stakhanova et al. taxonomy [167]. Papadaki et al. emphasize in [164] that intrusion response is not a straightforward process. In fact, in order to decide which response provides a better trade-off when reacting against an intrusion attempt, one must consider not only the intrusion impacts, but also the response impact on the target system and its users. One should thus guarantee higher response efficiency at a lower cost.

6.7.1 Intrusion Response/Countermeasure Taxonomies

Existing response taxonomies reflect different views regarding intrusion response mechanisms. Below we describe several existing taxonomies. We also briefly discuss the improvements they made and their limitations.

Few works have been done in the security response domain. Some authors have proposed countermeasure's taxonomies as a strategy to analyze and evaluate security mechanisms to mitigate intrusions and attacks.

Irvine and Lewis provide taxonomy [155] based on security goals (e.g. confidentiality, integrity, availability) that are used as a framework to define the costs associated to the network security services. However, there are some inconsistencies in the proposed taxonomy. For instance, data confidentiality as well as audit and intrusion detection is considered in the same group of criteria. The former is a security service, whereas the latter are security technologies.

Venter and Eloff propose taxonomy [170] for information security technologies that are used to secure information at application, host and network level. The taxonomy is classified in two main sections: proactive, which groups measures that are used as a preventive strategy (before the security breach occurrence); and reactive, which groups measures used as a response strategy (as soon as the security breach is detected). However, some concepts are ambiguous (e.g. access control and passwords are considered as reactive measures).

Wang and Wang present countermeasure taxonomy [171] based on the attack target (e.g. application layer, platform layer, and network layer) and it is categorized in 4 dimensions: standards and policies, library and tools, administration and system management, and physical tools. Authors present an evaluation of each security technology and its effectiveness in dealing with the applicable threats and risks. However, the taxonomy lacks of important concepts such as encryption, and some general concepts like biometrics are mixed with products such as Tripwire and SQLnet.

Schumacher, Kim et al., and Talib et al. have proposed the definition of a countermeasure taxonomy through the use of security ontology [165],[158],[168], in order to maintain a knowledge base of security patterns. However, most of the existing works lack of some concepts or do not clearly express the link between threats, assets and countermeasures. In addition, the definition of some concepts such as threat and attack remain ambiguous in some of the ontology while in others, such concepts are not even developed.

According to Thomas [169], there does not exist a response taxonomy that includes response strategy, timing response, and other factors related to actual relevance and efficiency of chosen countermeasures. In addition, the author suggests to distinguish between short-term response (countermeasures that only deal temporarily with threat e.g. server port filtering), and long-term response (countermeasures that stays active for long periods of time e.g. application security patches).

Fish taxonomy defined in [154] classifies intrusion responses according to their long term effects, by separating active from passive response mechanisms. While the former category includes responses that actively control intrusion damages and passivates their effects, the latter includes responses that only perform passive reaction such as alerting and reporting. This taxonomy is among the first response taxonomies ever proposed. It thus lacks interesting response properties such as response autonomy and response selection mechanism.

Carver et al. also provide intrusion response taxonomy in [149]. This taxonomy is attacker-centric, that is to focus on the attack mechanism and how response interleaves with this mechanism. It classifies responses according to the timing of the attack, that is preventive, reactive or recovery measures. It also classifies attacks according to their expected impacts on the target system. Meanwhile, this taxonomy does not consider response properties that are how a response prevents the intrusion success.

Attacker-centric taxonomies also include the Papadaki et al. taxonomy in [163] and the Killourhy et al. taxonomy in [157]. They both describe IT system intrusions and point out the attack properties that may be used for intrusion response. However, these are intrusion taxonomies rather than response taxonomies.

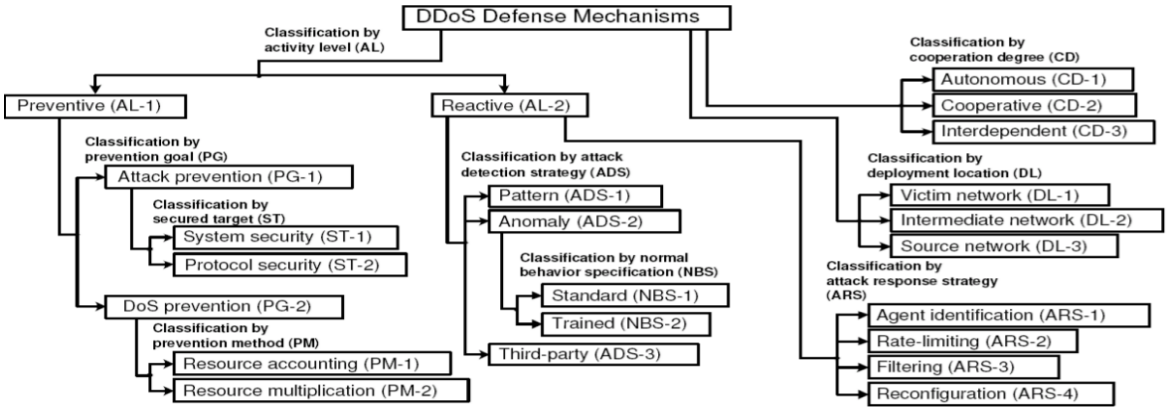


Figure 6-11 Mirkovic et al. DDoS response taxonomy

Mirkovic et al. provide in [159] a response taxonomy that considers only Distributed Denial of Service (DDoS) attacks. In fact, two separate taxonomies are provided in [159], discussing both DDoS attacks and their response mechanisms. The response taxonomy described in [159] covers interesting intrusion response mechanisms. Intrusion responses are classified into five categories: preventive measures, reactive measures (including DDoS detection), degree of cooperation, deployment location and response strategy. This is the only response taxonomy we found, and which classifies intrusion responses according to the intrinsic characteristics of the response mechanism. Meanwhile, it only covers distributed attacks against system availability.

Stakhanova et al. present in [167] taxonomy of intrusion response systems, together with a review of current trends in intrusion response. They discuss the properties of existing response methodologies, and conclude with a set of essential features as a requirement for an ideal intrusion response system. Stakhanova et al. classify response systems according to six criteria: activity of triggered response, degree of automation, ability to adjust, and time of response, cooperation ability and response selection method. The last four classification criteria are only relevant for automated response mechanisms. In contrast to attack-centric taxonomies in [149],[163],[157], Stakhanova et al. classify response systems from a system-centric perspective. They represent the response selection mechanism and the way it interleaves with the intrusion process. Unfortunately, this taxonomy does not consider intrusion impacts, nor the effect of responses as they reduce intrusion impacts. The remaining of this paragraph summarizes the response taxonomy in [167].

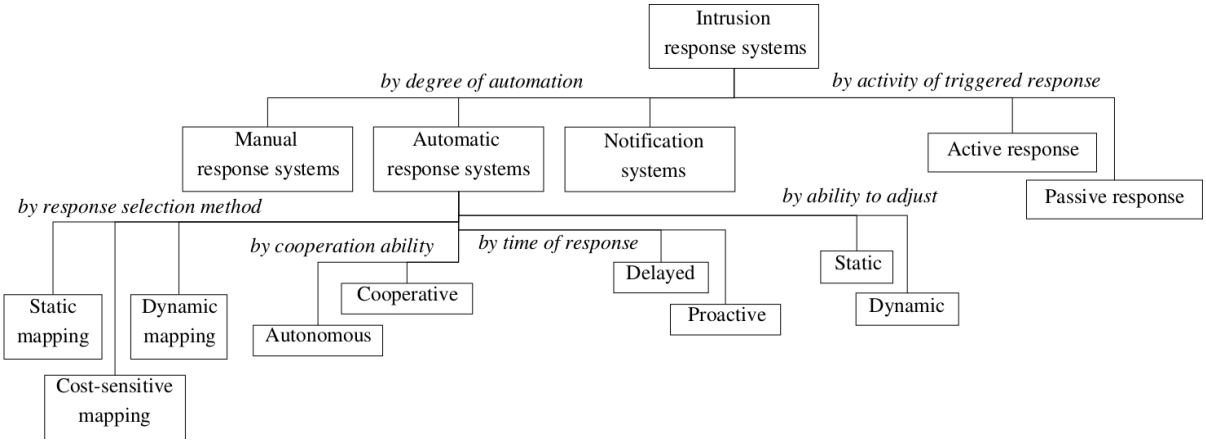


Figure 6-12 Stackanova et al. intrusion response taxonomy

Activity of triggered response

Characterizes the attitude of the response system when it detects an ongoing attack. It either passively or actively reacts against an intrusion attempt.

Degree of automation

Characterizes the tasks that are performed by the response system in order to assist the security operator. Three automation levels are described in [167]:

Ability to adjust

Expresses the ability of the intrusion response system to adapt its selected responses to the changing environment of an attack. Automatic response systems propose either static or adaptive intrusion response.

Time of response

Establishes a race condition between the intrusion attempt and the selected response. The efficiency of an intrusion response is strongly related to the response deployment time, which provides either proactive (also preemptive) or delayed (also reactive) response.

Cooperation ability

Describes the architecture of a response system and the data flow between its components. The Stakhanova et al. taxonomy points out two levels of cooperation: autonomous and cooperative response systems.

Response selection method

Describes the reasoning process and the level of sophistication that are used to implement the final response selection. Stakhanova et al. taxonomy distinguishes three response selection methods that are the static response, dynamic response and the cost-sensitive response. The static vs. dynamic response selection methods meet the classification by ability to adjust. We will thus discuss the remaining cost-sensitive response selection.

6.7.2 Policy-based response

Policy-based intrusion response automatically adjusts access control policies, either by strengthening or weakening access permissions, in order to enable the system to overcome an attack. We often refer to dynamic access control policy rules that are to constrain the activation of a policy rule to the satisfaction of environmental constraints. Activation constraints have different aspects according to the formalism that is adopted for the policy definition. In [166], it is claimed that by carefully setting policy activation constraints, one can use policies to handle intrusion prevention, detection and response.

Managing policy activation constraints for intrusion response requires interacting with both detection systems and policy enforcement points.

6.7.2.1 Policy-based response framework

In [151], authors use Or-BAC [156] contexts in order to implement policy-based intrusion responses. They specify mappings between alert attributes and concrete Or-BAC instances that are subjects, actions and objects. They propose architecture for a policy-based response system. This architecture implements a complete workflow that triggers new Or-BAC policy rules after intrusion alerts have been notified. These responses are represented as a set of policy rule instances.

6.7.3 Countermeasure Evaluation Methodologies

Research in the selection of appropriate countermeasures to mitigate the impacts of attacks is still in progress. Some authors have proposed qualitative methods (e.g. defence trees and conditional preference networks [147],[148], while others suggest quantitative methods (e.g. genetic algorithm [152], game theory [150],[153] that use cost sensitive metrics (explained in Section sec: models) to evaluate, rank and select countermeasures. This section details and classifies these methods into two approaches: qualitative and quantitative.

6.7.3.1 Qualitative Approaches

Qualitative approaches (e.g. defense trees and conditional preference networks [147],[148] have been proposed to evaluate and select countermeasures based on expert knowledge, organization's objective, and other useful criteria. The selection process does not generally rely on cost sensitive metrics, but they can use numerical data to decide upon several candidates. In addition, the degree of automation is generally low, meaning that in most cases, these are static methods that require the human intervention to select the countermeasure.

Defence Trees and Conditional Preference Networks

Bistarelli et al. [147],[148] propose an approach that uses two qualitative instruments for the selection of defense strategies to protect an IT system from the risk of attacks. The first approach is the use of defense trees to model attack/defense scenarios, and the second approach is the use of Conditional Preference Networks (CP-nets) to model qualitative conditional preference over attacks and countermeasures.

Defence trees are an extension of attack trees that represent an attack against a system and the way it can be mitigated by a set of countermeasures. The main difference between attack and defence trees is that the former represents only the action that an attacker can perform, while the latter adds the set of countermeasures that can be introduced into the system to mitigate the possible damages produced by an attack action.

Conditional preferences networks (CP-net for short) are a graphical formalism that specifies and represents qualitative conditional preference relations. CP-nets capture preference statements that are able to express a conditional preference over some variables. The following definition is proposed [2]:

> A CP-net is a directed graph N = (V,E), where V = $x_1, ..., x_n$ is a set of variables and E = $(x_i, x_j) : x_i, x_j \epsilon V$ is a set of edges between variables. The function $P_a(x)$ gives for each node x ϵ V , the node x' ϵ V s.t. (x', x) ϵ E. The conditional preference table of the CP-net describes a strict partial order $(D(x_i), \succ_i^u)$ where $D(x_i)$ is the domain of the variable x_i, and $\succ_i^u$ represents the conditional preference of the instantiations of variable x_i given an instantiation u of the variable $P_a(x_i)$.

The conditional preference table is specified by the system administrator based on expert knowledge and statistical information. As a result, it is possible to determine, in a qualitative manner, the attack strategies that an attacker may follow to damage a system, the different actions that compose each attack and the countermeasures that a system administrator can implement on the system.

Multi-objective Selection

The multi-objective countermeasure selection approach proposed by Neubauer et al. [160],[161] is a workshop that provides a structured and repeatable process that includes the following steps:

- Evaluation criteria according to the organization's strategy
- Assessment of the existing IT security infrastructure
- Identification of Stakeholder preferences
- Determination of the solution space of all efficient countermeasures
- Selection of the individual best countermeasure

In addition, the approach takes into account interdependencies among security countermeasures and provides an environment for multiple users. A moderator is required to provide advance and professional support during the workshop and the interactive selection allows decision makers to playfully explore the alternative that matches their preferences. The workshop is divided into two sections that are performed in a full-day meeting. The first section consists on the assessment of the existing countermeasure portfolios and the subsequent generation of promising solutions. The second section pretends to iteratively reduce the number of portfolios until identifying the portfolio that best fits the stakeholder's objective.

The portfolio presents the different countermeasures or combinations of them according to multiple criteria (e.g. monetary value, accept cost, setup costs, setup time, etc.) As a result, the approach serves as a valuable tool to improve security awareness of top management as they may run through different scenarios and potential solutions which should decrease the probability to overlook relevant risks.

Threat Tree

Bedi et al. [146] propose an approach that uses threat tree to select optimal countermeasures. Threat modeling involves understanding adversaries' goals in attacking a system based on system's assets of interest. The process consists of decomposing the application, identifying, ranking, and mitigating threats. In order to identify threats it is necessary to go through each of the security critical entities and creating threat hypotheses that violate confidentiality, integrity, or availability of the entity. Threat trees are then designed for each threat requiring mitigation to analyze the threat through attack paths. The root node of a tree is the threat, each leaf node is an attack to accomplish the threat and the path from leaf to root is the way an attacker achieves the threat, as depicted in Figure 3.

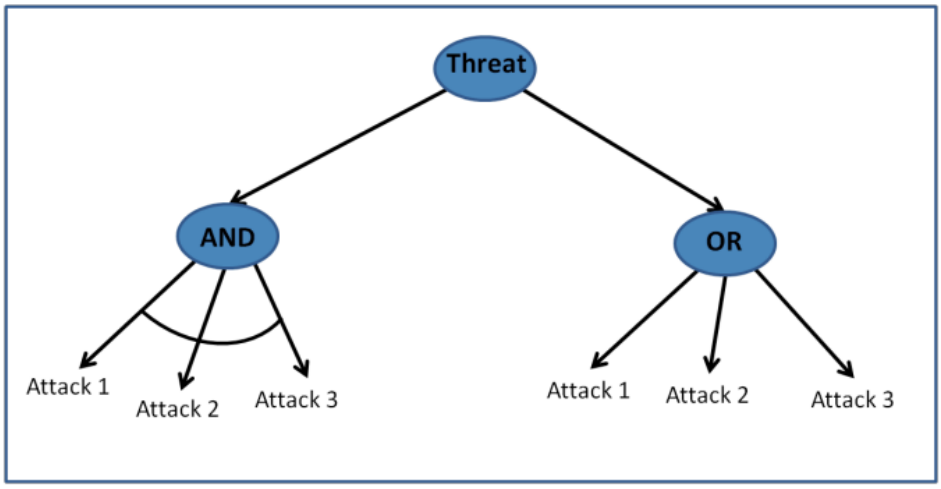


Figure 6-13 Bedi et al. Threat Tree

In Figure 6.13, the AND refinement means that in order to occur the root threat, all the corresponding attack must occur, whereas, the OR refinement means that in order to occur the root threat, at least one attack should occur.

The approach to select optimal countermeasures needs to be adopted at the design phase of software life-cycle. The threat that needs to be mitigated is at the root node. The attacks at leaf nodes cannot be refined further as they are executed by the attacker to accomplish the threat. Countermeasures are therefore applied against the attacks at leaf nodes to prune the attack branches from the threat tree to avoid the threat at the root level.

The solution has been designed to generate a multi-threat attack graph by combining all the individual threats responsible for the security compromise of the system and removing duplicate nodes in multiple threat trees. This graph gives a unique set of attacks requiring mitigation as output. In addition, the solution prioritizes the identified attacks for mitigation on the basis of frequency and the expected damage the threats can generate to the system. Some of the attacks having threat_index less than the associated threshold value are ignored for mitigation, making this approach economical for software security. As a result, the mechanisms are proven to optimally save the system from being compromised.

6.7.3.2 Quantitative Approaches

Quantitative approach consists of those studies in which the data concerned is analyzed in terms of numbers. Quantitative methods (e.g. genetic algorithm [152], game theory [150],[153] generally use one or several cost sensitive metrics (explained in Section sec:models) to perform the evaluation and selection of countermeasures. They are typically dynamic, and the degree of automation is higher than the one presented in qualitative methods.

Game Theory

Cavusoglu et al. [150] and Ferenc et al.[153] have used the Game theory logic to evaluate and select countermeasures for a given attack. Game theory is used to analyze problems in which the payoffs to players depend on the interaction between players' strategies. The analogy in the IT security investment environment is that firms and hackers are players. The firm's payoff from security investment depends on the extent of hacking it is subjected to. The hacker's payoff from hacking depends on the likelihood of being caught, which, in turn, depends on the level of investment the firm makes in IT security. The first step in using game theory to analyze such strategic interactions among players is to develop a game tree that depicts the strategies of players.

The game starts by selecting the type of traffic to the system, which can be external (with probability ε) or internal (with probability $1-\varepsilon$). A given node represents external users that can be either authorized or unauthorized. Similarly, one node characterizes internal users that can be either honest or dishonest. A dishonest user can take two actions: hack or not to hack. If the hacker decides to hack, the game moves to the following node. The firm makes decisions about whether or not to monitor based on the state (signal or no-signal). The firm must make decisions without knowing exactly which node the game has reached. However, it can determine the probability of intrusion in the signal and no signal states using Bayes Rule as illustrated in Equation 1.

$$P(i,s) = \frac{P(s,i)P(i)}{P(s,i)P(i)+P(s,\bar{i})P(\bar{i})}$$
$$P(i,\bar{s}) = \frac{P(\bar{s},i)P(i)}{P(\bar{s},i)P(i)+P(\bar{s},\bar{i})P(\bar{i})} \quad (1)$$

Where

$P(i,s)$ Probability of intrusion given signal,

$P(i,\bar{s})$ Probability of intrusion given no-signal,

$P(i)$ Probability of intrusion

$P(s)$ Probability of signal

$P(s,i)$ Probability of signal given intrusion

$P(\bar{s},i)$ Probability of no-signal given intrusion,

$P(\bar{s}, \bar{i})$ Probability of no-signal given no-intrusion

As a result, the model can be used in a variety of ways, for instance, it can be used to select a specific security measure. Similarly, the model is used as a what-if analysis tool to explore different options and evaluate the effect of a given parameter in the countermeasure selection.

Genetic Algorithms

Genetic Algorithms have been shown to work in many large complex search problems with affordable space and time requirement [152]. The idea of this approach is inspired on a biological metaphor that searching could be viewed as a competition among a population of evolving candidate problem solutions (represented by chromosomes). Therefore, through operations analogous to gene transfer in sexual reproduction, solutions from one population are taken and used to form a new population by means of operators such as crossover and mutation. A fitness function evaluates each solution to decide whether it is capable of contributing to the next generation. The underlying hope is new population is better than the old one, thus sufficient evolution would ultimately lead to an optimized solution.

The approach evaluates the performance of selected security measures through Equation 2

$$NPV = Cost^{dev} + (ALS - Cost^{op}) \times \sum_{i=1}^{n} \frac{1}{(1+r)^{i-1}} \quad (2)$$

Where

NPV represents the Net Present Value,

$Cost^{dev}$ refers to development costs,

ALS is the Annual Loss Savings that results of deploying a set of countermeasures,

$Cost^{op}$ refers to operational costs,

n is the number of periods under consideration,

r represents the discount rate

The genetic algorithm approach allows specifying an upper limit on acceptable unmitigated ALE; and the countermeasure failure, to find a solution that provides enough protection even under failure conditions. As a result, the genetic algorithm is able to find the best countermeasure combination in all studied cases.

Decision Matrix

Whenever several solutions are presented to mitigate a given threat or attack, a decision must be taken to select the most convenient countermeasure, whether the organization prefers the solution with the lowest cost or the most effective one, it is not always easy to reach a consensus. Norman [162] proposes a decision matrix to help security administrators in deciding upon the most appropriate countermeasure to implement. This tool was designed to prevent terrorism attacks and other physical attacks, but it can be extended to other domains such as the Information Technology. The decision matrix lays out the goals, risks, costs, and several other factors. In addition, countermeasures are scored based on their costs, their ability to achieve goals, and to mitigate threats, as depicted in Figure 4.

Countermeasure Methods	Goals Achieved						Risk Mitigated					Score	Rank	Accepted Risk					Estimated Cost	Effectiveness	Convenience
	1	2	3	4	5	6	A	B	C	D	E			A	B	C	D	E			
Fence entire property	1			1	1		1	1	1	1		7	2					x	400,000	High	High
Fence parking lots and garage			1	1	1		1					4	3		x	x	x	x	100,000	Low	High
Use landscaping to create a barrier				1	1	1						3	4	x	x	x	x	x	400,000	Low	High
Use landscaping and fencing to enclose property and deny access	1	1	1	1	1	1	1	1	1	1		10	1					x	500,000	High	High

Goals Description

1. To deny access to unauthorized persons
2. To create a pleasant and visually pleasing environment
3. Cost effectiveness based on goals and threats mitigated or eliminated
4. Convenience for employees
5. Conformance to business culture
6. Conformance to aesthetic values

Risk Description

A. Harmless unexpected visitor
B. Unauthorized external visitor
C. Property criminal
D. Personal or sexual attack criminal
E. Workplace violence visitor

Figure 6-14 Norman T. Decision Matrix

The matrix presented in Figure 6.14 shows the risks an organization is willing to accept if a given countermeasure is selected. The construction of this matrix starts by listing the goals of the countermeasures (e.g. access control, deterrence, detection, assessment, delay), numbered from 1 to N. This is followed by listing the possible risks for the countermeasure to mitigate (e.g. confidentiality, integrity, availability), lettered from A to X. Then, we list the countermeasure methods in rows as well as goals achieved, score, rank, accepted risks, estimated costs, effectiveness and convenience. As a result, security analysts are able to select, optimal countermeasures based on a multi-factor evaluation matrix.

It is important to note that we can add as many goals and risks as possible to evaluate the different countermeasures. Scores are based on highest number of goals achieved and risks mitigated or eliminated. The ranking column is based on the highest score, and the columns for costs, effectiveness and convenience are estimations based on expert knowledge.

7. Simulation

This section covers the main idea of simulation of DDoS attacks and data features to be used in simulations. There are few works on building a general DDoS defense simulation and evaluation methodology, in the literature. Simulation of DDoS attacks is a challenging work, since so many different parameters have to be included in the design of the test bed. In the next section a brief explanation of challenges of DDoS simulation will be given. There are also various methods proposed so far for detection of DDoS attacks. These methods use different features of network data. A general coverage of these methods and datasets used in these methods will be given in the features section. Finally, a brief explanation about future steps in simulation of DDoS attacks is given in the last section.

7.1 DDoS attacks simulation

DDoS attacks are multidimensional processes. In simulation and recreation of DDoS attacks, there are multiple tasks to be carefully designed. The methodology must provide detailed steps that guide us in experimentation and defining evaluation. In [15], five main tasks are given that an experimenter needs to design. These are:

1. Attack mechanism
2. Background traffic
3. Network topology
4. Defense mechanism
5. Measurement and metrics

7.1.1 Attack Mechanism

There are multiple ways for denying a service. We have to obtain tools to realize these attacks. It can be possible to obtain real DDoS software from underground networks, but these tools are more complicated in controlling agent machines than launching attacks.
It will be time waste to use these real attack tools. Since defense mechanisms will be tested software written for research, they will be more useful. These kinds of tools are available in site http://www.isi.edu/deter/tools.html.

7.1.2 Background Traffic

Background traffic modeling is another step in evaluating a defense mechanism. Different results can be obtained in the same testing methodology by using different background traffic. The simplest form of background traffic generation is using packet trace replay [204]. Using multiple PC's to replay real packet traces through high data transmission speeds can allow us obtain high-speed background traffic. Since many defense systems need to be tested under realistic traffic conditions, packet trace replay seems to be suitable to our needs. However, re-playing same packet traces can result same statistical behavior looking into details of the packet in background traffic. Real background traffic is rather having many different types of packet traces. Another approach in background traffic generation is to use application-specific traffic generators. These tools model network traffic based on different applications such as ftp and http. Some of them are Surge [173], Trafgen [190] and PackMime [176]. By using different kinds of application Specific traffic generator tools, we can obtain more complicated background traffic. There are also application independent traffic generators that create traffic at IP level. They create network traffic based on probabilistic distributions and stochastic processes for various traffic parameters such as inter-packet gap interval and packet size.

Evaluation Methods develop a collection of these generators for Internet Security Technologies (EMIST) DDoS [194]. It includes configuring a wide mix of background traffic that consists of TCP traffic created using Harpoon [201], DNS traffic by setting up a server and periodically issuing requests from various locations in the topology, and ICMP echo request and reply traffic using the ping utility.
All the listed software will be examined in details and used in simulation environment.

Defense mechanism: there is a large number of DDoS defense mechanisms developed so far.
These defense mechanisms vary greatly in their approaches to attacks. Some systems aim only detecting the attacks, some of them attempt to mitigate under attack, some others try to filter attack traffic while protecting legitimate usage. To thoroughly evaluate a defense, one must be aware of its approaches to attack detection, response, prevention or trace back, and stress test them by generating attacks that attempt to bypass or crash the defense. In [196] list recommended test scenarios for some general defense categories are given. These scenarios are:

- Defenses that learn the difference between the legitimate traffic should be tested with flooding
- Attacks that mimic legitimate traffic features and slowly increase their rate to achieve values that deny service.
- Defenses that use resource accounting should be tested with highly distributed attacks
- Interdependent defenses should be challenged with attacks on the defense itself, and in presence of control message loss, to evaluate whether defense modules can function when isolated from their peers.
- Defenses that perform agent identification should be tested in topologies that have high levels of path sharing among legitimate users and attackers, and with highly distributed attacks where each agent floods at a low packet rate.
- Defenses that detect attacks and respond to them in some fashion should be tested with short-duration, repetitive attacks to evaluate the cost of turning the defense on and off and the overall protection offered to the attack victim.
- Defenses that deploy some kind of cooperative defense should be tested for insider attacks to evaluate the damage that a trusted member could indict to a system, if compromised by an attacker.

Above scenarios will be used for evaluation of defense mechanisms. For all kind of defense mechanisms, a plan will be constituted before making simulations.

Measurement and metrics: A proper metric for evaluation of defenses should be decided. Generally, used metrics for performance evaluation of intrusion detection systems are ROC (Receiver Operational Characteristics) curves and CID (Intrusion Detection Capability). However, there is lack of standard metrics that can be used for evaluation of DDoS defenses. We have to build our own metrics assessing countermeasure effectiveness. One alternative is to determine the percentage of blocked attack traffic compared to the percentage of allowed legitimate traffic.
One of the most challenging parts of the evaluation methodology is to use a proper measurement metric. There is a lack of a standard set of metrics that can be used to evaluate a mix of DDoS attacks and defenses in various experiments. There are individual research efforts and commercial products utilized a variety of metrics to measure and assess the results of their respective techniques, products and technologies [195]. In [200] performance metrics is divided into two categories, namely extrinsic and intrinsic. Extrinsic metrics are measures that can be computed and observed by external parties in relation to the object (attack, defense etc.) being measured. On the other hand, intrinsic metrics can only be computed by the object being measured and only by analyzing the internal algorithms and data structures such as queues and connection tables.

Different intrusion detection systems can be used for detection of attacks. These systems will return a probability of attack. The reason of using multiple methods for detection process is to make a robust decision system. By simulating DDoS attacks in a controlled environment, we will have enough data for evolution of different intrusion detection systems. The simulations provide data to compare these IDSs considering metrics such as false positives, false negatives and intrusion detection capability (CID). Although we will use intrusion detection systems decision of DDoS attack existence will be performed by a decision mechanism that uses the outputs of the intrusion detection systems. This decision process can be simply a weighted sum of the attack probabilities or a Bayesian network including probability of detection of these intrusion detection systems.

Related Work: There is limited work on simulation of DDoS attacks in the literature. In [195] a common evaluation methodology for distributed denial-of-service (DDoS) defenses is proposed. The proposed methodology can be used and enables independent evaluation and comparison of DDoS defenses. They gave a set of automated tools to harvest typical attack, legitimate traffic and topology samples from internet. Their work consists of a benchmark suite defining the elements necessary to recreate DDoS attack scenarios in a test bed setting.

- A set of performance metrics that express a defense system's effectiveness, cost, and security
- A specification of a testing methodology that provides guidelines on using benchmarks and summarizing and interpreting performance measures.

7.1.3 DETER test bed

Rapid advances are urgently needed to defend against network attacks such as distributed denials of service, worms, and viruses. These cyber-security problems include some of strategic importance, like the protection of critical infrastructure. Rapid advances require an improvement in the state of the art of experimental evaluation of network security mechanisms.

Such efforts require the development of large-scale security test beds [184], combined with new frameworks and standards for testing and benchmarking to make the test beds truly useful. Current impediments to evaluating network security mechanisms include lack of scientific rigor [199]; lack of relevant and representative network data [183]; inadequate models of defense mechanisms; and inadequate models of the network, background, and attack traffic data[180]. The latter is challenging because of the complexity of interactions among traffic, topology, and protocols [189], [190].

In addition to the hardware and software infrastructure needed to conduct experiments, the DETER test bed provides tools that aid the experimenters, many of which are being developed by experimenters themselves. The public Internet must be protected from the side effects of the security experiments that run on the test bed and the experiments must be protected from interference from the Internet.

Cyber-defense research has been severely limited by the lack of a public experimental infrastructure for testing new theories and new technologies in realistic scenarios. It is both unclear and unproven that technologies tested on small subnet-sized topologies modeled by a few machines will scale to realistic Internet environments. To meet this challenge, the cyber-Defense Technology Experimental Research (DETER) test bed has been developed. The DETER test bed is intended to provide an experimental infrastructure to support the development and demonstration of next-generation information security technologies. DETER provides a medium-scale facility for safe, repeatable security-related experimentation, to validate theory and simulation. The DETER test bed is implemented as an Emulab [202] cluster, using the comprehensive and powerful cluster test bed control package developed by Jay Lepreau and his colleagues at the University of Utah. With a current design point of several hundred experimental nodes, the DETER test bed pro-vides an intermediate point between small-scale and Internet-scale experiments. Since it is chartered to support scientific investigation, the test bed is designed with experimental repeatability as a fundamental requirement. Repeatability allows experimenters to deeply investigate, validate, and find alternative explanations for their research results and to build upon the results of others.

The DETER test bed must provide containment of malicious code as well as control over the effects of generated attack and background traffic. Because different experiments pose different levels of threat to the public Internet, it is important to balance on a case-by-case basis the cost and complication added by isolation with the level of threat posed by the individual experiment. A simple test bed can be constructed by manually wiring together and configuring a dedicated set of machines; however, such a test bed lacks generality and sharability. DETER (like Emulab) belongs to the more useful class of test beds that are general-purpose and support remote access. It can be used effectively for a wide variety of experiments, and an experimenter can reconfigure and control experiments remotely. Additionally, DETER is partitionable into multiple independent experimental test beds that can be used simultaneously, allowing more efficient use of its hardware resources. For all but the most dangerous experiments, the test bed must be remotely accessible to experimenters for initiation and monitoring. An experimenter must be able to control the experiment even when the test network is congested or broken as the result of the experiments that are run.

The requirement for remote accessibility may clash with security and containment requirements.
The test bed must also be sharable in time and space among a large community of users, while providing strong isolation of effects among users (e.g. traffic, attacks, etc.). Just as a major particle accelerator needs to have multiple beam-lines, so the DETER test bed needs to support multiple simultaneous experiments.

The DETER test bed has been operational since March of 2004 and is used by researchers to perform experiments on worm propagation, distributed denial of service attacks, and routing and infrastructure attacks. At the time of writing, the test bed had 231 nodes and it has been used by commercial and academic researchers to study attacks and assess the benefit of products in development.

The test bed provides investigators with the ability to run experiments using potentially risky code, on an isolated experimental network. For most categories of experiments, control is possible remotely by connecting to a test bed user machine through the Internet. Firewalls, intrusion detection systems to monitor access, and other safeguards protect access through this control network, and physical separation from the Internet is provided on the experimental network on which the experimental nodes communicate. The test bed provides a focus of activity for a community of academic, industry, and government researchers. Regular meetings of the user community provides an opportunity for investigators to show early results, and to help one another in the use of the DETER test bed.

Support for test bed users includes a repository of attack traffic generators, monitoring tools, topology generators, and other tools, and work is underway to integrate these tools into an experimenter's workbench, which will simplify the task of getting new experiments up and running.

7.1.4 Matching pursuit anomaly detection

7.1.4.1 Data Collection Mechanisms

The study of the data collection mechanisms is important because the detection performed by an intrusion detection system can only be as good (in terms of accuracy, reliability and efficiency) as the data on which it bases its decisions. If the data are acquired with a significant delay, detection could be performed too late to be useful. If the data are incomplete, detection abilities could be degraded. Moreover, if the data are incorrect (due to error or to the actions of an intruder), the intrusion detection system could stop detecting certain intrusions, giving its users a false sense of security. Unfortunately, these problems have been identified in existing products.

In general, data collection methods can be grouped into two main categories; network based data collection and host based data collection. These categories are explained in subsequent paragraphs.

Network Based Data Collection

Network traffic information is collected from a typical network sniffer tool. These data are the most demanding due to its sheer volume. Consequently, statistical techniques are generally required to sample the network activity. This can miss information on the network but should capture ongoing activity, even if intermittent. Capturing network traffic data will often lead to surprising results as to what activity is occurring on the network and has frequently been used to identify intrusions. Each packet on the network, whether TCP/IP or UDP based, provides header information identifying the type of data being transmitted, the originating system, and the destination system. The type of data being transmitted will directly implicate the tool used to generate the data. This has been used numerous times to identify illegal IRC servers, most often being run on compromised systems.

It is also used to identify systems illegally connected to the local sub network and compromised systems no longer providing complete or correct log facilities.

Host Based Data Collection

Most of the intrusions that existing intrusion detection systems can detect are caused by actions performed in a host: executing a command, gaining access to a service and providing it improper data, etc. The attacks act on the end host, although they may occur over a network. The only attacks that act on the network itself are those that flood the network to its capacity, preventing legitimate packets from wing. However, we claim that most of these attacks can also be detected at the end hosts. For example, a ping flood could be detected at the ICMP layer in the host by looking for the occurrence of a large number of ECHO REQUEST packets.

As it is installed on a host, data collector can monitor system resources as well as look at operating system audit trails or application logs. It is also independent of the network speed as it monitors only a single host. However, the system administrator now needs to install a number of monitors instead of just one, thus incurring more administrative overhead. In addition, the user could experience a performance penalty as the monitor is on the same host as the application. Furthermore, most monitors on the OS level cannot detect attacks directed at the lower network protocol levels because network information typically does not become available in the audit event stream until it has reached the higher protocol levels.

7.1.4.2 Features

There are plenty of data features that can be used in detection of DDoS attacks. Some examples include traffic flow information [197], [189] router SNMP MIB variables [205], TCP and ICMP header information [180]. The main challenge is to select most convenient features from available dataset for achieving optimal detection performance.

7.1.4.3 IDS Features

Features used in prior research on IDSes can be organized into many categories. Next sections will include three of these categories; *Flow based features*, *packet based features* and *SNMP based features*.

Flow-based Features

Lakhina et al. [188] analyzed events that affected to the distribution of traffic features and marked these as anomalies. They monitored network-wide backbone traffic using the following IP packet header data:

- Source IP address
- Destination IP address
- Source port number
- Destination port number.

They grouped known anomalies into seven categories based on the type of the detected attack.
These were DoS, Flash Crowd, port scan, network scan, outage events and worms to name few.
The classification was done using multiway subspace method together with the k-means clustering algorithm. The multiway subspace method is able to isolate correlated changes on the four IP packet header features (source and destination IP address, source and destination port number) among traffic flows [188].
The same features are also used by Fontugne et al. [181] in their image processing based approach to detect anomalies. They compared their proposed anomaly detection method against a statistical-based method proposed by Dewaele et al. [178]. The comparison was done using a network traffic data collected from Trans-Pacific. Fontugne et al. [181] categorized the results in similar way than Lakhina et al. [188] did but instead of grouping the detected anomalies into seven groups, they grouped them into 15.

Gorton [182] used two detection methods to analyze a router log data; a single event and threshold analysis. The single event analysis raises a g of intrusive activity when a single event is discovered.
In the threshold analysis, intrusive activity is flagged with respect to accumulated activities. In his analysis, he collected syslog messages from Cisco routers and transformed the log data into a set of features that are:

- Time from the syslog
- Status that can be either permitted or deny
- Protocol identifier
- Type of service
- Source IP address
- Source port number
- Destination IP address
- Destination port number
- Number of ICMP messages
- Number of packets.

With single event analysis Gorton was able to detect spoofed connection attempts, connection attempts to known Trojan horses, connection attempts to known vulnerable ports, the Land DoS attack, TCP-broadcasting, the echo-chargen attack, ICMP and UDP echo request. With threshold analysis, Gorton was able to detect SYN flooding, network mapping and port scans to name few.

Knuuti [198] compared the usability and performance of three different IDSes in a large IP networks. The evaluated IDSes were Snort, Bro-IDS and TRCNetAD. Snort and Bro-IDS are capable of analyzing traffic in real-time when TRCNetAD is a non-real-time anomaly detection based IDS.[198] Features that Knuuti used are:

- IP address
- Time stamp
- Number of ICMP packets
- Number of UDP flows
- Number of TCP connections
- Amount of received data
- Amount of sent data
- Number of received packets
- Number of sent packets
- Number of different port numbers used over 1024
- Number of port numbers used over 1024
- Number of different port numbers used below 1024

- Number of port numbers used below 1024
- Number of receiving sequences from different IPs
- Number of receiving sequences
- Number of sending sequences to different IPs
- Number of sending sequences.

Knuuti conducted two, one week long, traffic capturing periods to collect data for the IDSes.
From the data collected he then generated time series that are 60 minutes long in order to create clusters and analyze the data with self-organizing maps. Snort detected over 1.5 million intrusions during the one-week traffic-capturing period. Snort was able to detect the following attacks:

- Buffer over flow attacks
- Trojan
- Denial of service
- VoIP attacks
- Heap over flow attack
- DNS spoofing attack
- Spyware.

Bro-IDS detected approximately eight thousand intrusions, which were, address and port scan.
TRCNetAD detected 150 thousand anomalies during the same time period. Knuuti also evaluated alarm similarities among the detectors and his conclusions were that TRCNetAD was able to detect some of the port and address scans that Bro-IDS discovered but there were no similarities among Snorts and TRCNetADs findings.

Packet-based Features
Kabiri et al. [187] have conducted research on identifying effective features for intrusion detection. They have done related research for detecting probing attacks [206] and for detecting smurf attacks [177]. Results from these researches are used in [187] as well.

Kabiri et al. [187] used Lincoln laboratory dataset 1998 to select optimal features from the IP and TCP packet header fields. Appendix 1 lists all the 32 basic features that they extracted from network traffic header fields. They used principal component analysis (PCA) method to select optimal feature subsets from the 32 features for each of the five categories in the Lincoln laboratory dataset. The suggested feature subsets are listed in Table 7.1 In their work Kabiri et al. [187] investigated the information value for each category and their conclusion for future work stated that these features should be experimented in an intrusion detection system. In addition a comparison of accuracy and efficiency should be done using the feature subsets and by using all the 32 features.

Carrascal et al. [177] used self-organizing maps together with learning vector quantization in their machine-learning based method to detect intrusions. They evaluated their anomaly detection efficiency by using Lincoln laboratory data sets as a testing data. Their systems detection rate was 72% and false positive rate 2%.

In comparison they provided a list of other AD methods whose detection rate was better than their methods but with a higher false positive rate. Features that Carrascal et al. used were:

- Codification of TCP flags
- IP protocol number
- IP type of service
- TCP window
- Packet size
- Codification of <source port / source IP address, destination port / destination IP address>
- Destination port
- Source port
- Source IP
- Destination IP
- Codification of TCP options.

Most of the features are self-explanatory but the coded features are not as clear. Carrascal et al. combined features that have multiple parameters such as TCP flags and TCP options into single features. The authors do not explain in details how the codification is done so one can only guess what the exact features are in reality.

SNMP-based Features
Lee et al. [175] used Simple Network Management Protocols Management Information Base (SNMP MIB) to detect intrusion. SNMP is a protocol used in TCP/IP network management and the idea to use it as a security-monitoring tool is intriguing. SNMP logs are generated in network devices in any case and by using, the already available logs do not add new requirements to the network infrastructure. By using SNMP MIB, some of the challenges in network intrusion detection can be avoided. There are no privacy concerns, as user confidential information is not needed for the analysis. In addition, the data rates are low compared to network traffic amounts. SNMP MIB does not require any new hardware as the SNMP is widely supported.

In their work, Lee et al. [175] used 12 features from SNMP MIB in intrusion detection. Traffic on interfaces is estimated by analyzing the correlation between IP group objects and interface group objects of SNMP MIB. In conclusions, they proposed that only IP group features could be used to enhance the analysis performance.

7.1.4.4 KDD99 Dataset

The 1998 DARPA Intrusion Detection Evaluation Program was prepared and managed by MIT Lincoln Labs. The objective was to survey and evaluate research in intrusion detection. A standard set of data to be audited, which includes a wide variety of intrusions simulated in a military network environment, was provided. The 1999 KDD99 intrusion detection contest uses a version of this dataset. Lincoln Labs set up an environment to acquire nine weeks of raw TCP dump data for a local-area network (LAN) simulating a typical U.S. Air Force LAN. They operated the LAN as if it were a true Air Force environment, but peppered it with multiple attacks.

The raw training data was about four gigabytes of compressed binary TCP dump data from seven weeks of network traffic. This was processed into about five million connection records. Similarly, the two weeks of test data yielded around two million connection records. A connection is a sequence of TCP packets starting and ending at some well-defined times, between which data flows to and from a source IP address to a target IP address under some well-defined protocol. Each connection is labeled as either normal, or as an attack, with exactly one specific attack type. Each connection record consists of about 100 bytes.

Since 1999, KDD99 has been the most widely used data set for the evaluation of anomaly detection methods [192]. This data set is built based on the data captured in DARPA'98 IDS evaluation program [191]. DARPA'98 is about 4 gigabytes of compressed raw (binary) tcpdump data of 7 weeks of network traffic, which can be processed into about 5 million connection records, each with about 100 bytes. The two weeks of test data have around 2 million connection records.

KDD99 training dataset consists of approximately 4,900,000 single connection vectors each of which contains 41 features and is labeled as either normal or an attack, with exactly one specific attack type. The simulated attacks fall in one of the following four categories:

1. *Denial of Service Attack (DoS):* is an attack in which the attacker makes some computing or memory resource too busy or too full to handle legitimate requests, or denies legitimate users access to a machine.
2. *User to Root Attack (U2R):* is a class of exploit in which the attacker starts out with access to a normal user account on the system (perhaps gained by sniffing passwords, a dictionary attack, or social engineering) and is able to exploit some vulnerability to gain root access to the system.
3. *Remote to Local Attack (R2L):* occurs when an attacker who has the ability to send packets to a machine over a network but who does not have an account on that machine exploits some vulnerability to gain local access as a user of that machine.
4. *Probing Attack:* is an attempt to gather information about a network of computers for the apparent purpose of circumventing its security controls.

It is important to note that the test data are not from the same probability distribution as the training data, and it includes specific attack types not in the training data, which make the task more realistic. Some intrusion experts believe that most novel attacks are variants of known attacks and the signature of known attacks can be sufficient to catch novel variants. The datasets contain a total number of 24 training attack types, with an additional 14 types in the test data only.

KDD99 features can be classified into three groups:

- *Basic features:* This category encapsulates all the attributes that can be extracted from a TCP/IP connection. Most of these features leading to an implicit delay in detection. Traffic features: This category includes features that are computed with respect to a window interval and is divided into two groups:
- *"Same host" features:* examine only the connections in the past 2 seconds that have the same destination host as the current connection, and calculate statistics related to protocol behavior, service, etc.
- *"Same service" features:* examine only the connections in the past 2 seconds that have the same service as the current connection.

The two aforementioned types of "traffic" features are called *time-based*. However, there are several slow probing attacks that scan the hosts (or ports) using a much larger time interval than 2 seconds, for example, one in every minute. As a result, these attacks do not produce intrusion patterns with a time window of 2 seconds. To solve this problem, the "same host" and "same service" features are recalculated but based on the connection window of 100 connections rather than a time window of 2 seconds. These features are called *connection-based traffic features*.

- *Content features:* Unlike most of the DoS and Probing attacks, the R2L and U2R attacks do not have any frequent sequential pattern. This is because the DoS and Probing attacks involve many connections to some host(s) in a very short period of time; however the R2L and U2R attacks are embedded in the data portions of the packets, and normally involves only a single connection.

To detect these kinds of attacks, we need some features to be able to look for suspicious behavior in the data portion, e.g., number of failed login attempts. These features are called content features.
Subsets of KDD99 Dataset: KDD99 is actually composed of three datasets. The largest one is called Whole KDD, which contains about 4 million registers. This is the original dataset created out of the data collected by the Sniffer.

Since the amount of data to be processed is too high, it is interesting to reduce the computational costs involved as much as possible. Thus, a subset containing only 10% of the training data, taken randomly from the original dataset was created. This resulted in the 10% KDD dataset used to train the IDS. It contains more examples of attacks than normal connections and the attack types are not represented equally. The list of class labels for 10% KDD is detailed. Because we are just interested in DoS attacks, we have listed attacks corresponding to DoS attacks.

In addition to the 10% KDD and Whole KDD, there is a testing dataset known as Corrected KDD. This dataset does not have the same distribution of probability of attacks, as is the case in the other bases. This happens because the Corrected KDD includes 14 new types of attacks aiming at checking the IDS performance to unknown forms of attacks. Note that in the complete dataset (Whole KDD) and in the training dataset (10% KDD) there are 22 types of attacks in total. It is also important to mention that the KDDs training dataset contains a large number of connections for the categories normal, probe and DoS. They represent approximately 99.76% of the whole dataset.

7.1.4.5 Feature Reduction

In the feature reduction method, a new set of features is extracted based on the features available from the data monitored such as network traffic data. The basic idea behind feature reduction method is to reduce the total number of features used in the network traffic model training. In general, feature reduction means that during a certain period of time a number of different features are monitored and a new set of features are then calculated from this monitored data. For example, the feature reduction tool could monitor number of packets to a specific destination, within a certain period. Then, once the monitoring period is over, a new feature (number of packets to that destination) is available for the IDS.

Another example of a feature reduction method is a principal component analysis (PCA). PCA is an algorithm that checks and converts the data set for all the correlated variables into a set of uncorrelated variables, also known as principal components. [203] KDD99 dataset can be thought as an example of feature reduction. The KDD99 consists of features that are calculated from the network packet-based traffic in the Lincoln laboratory dataset to aow-based traffic. These converted features are used for example in machine-learning based IDSes.

Challenges in Feature Extraction Scalability are an issue with IDSes. Because of the huge amount of data owing through the network, it is not an easy task to find out the right information needed for an IDS. The problem is to find an answer for the question: What features need to be taken into account when calculating or analyzing whether the activity is malicious or not? In telecommunications networks link traffic can reach up to 150 Gbps traffic rates while current IDSes are capable of monitoring only some parts of the traffic. For example, Source res IPS is capable to monitor network traffic speeds from 5Mbps up to 20 Gbps [185]. In order to cover the whole bandwidth, the traffic needs to be divided somehow and monitored by multiple IDSes. Then again, the information provided by the IDSes needs to be correlated somehow which again adds another challenges to the whole intrusion and anomaly detection process.

Based on prior research on IDSes it is clear that either one of the techniques alone cannot detect everything but the combination of the both is the most promising approach. For example, misuse detection can be used to filter known threats from the traffic to make it easier for the anomaly detection system to focus on the unknown. Even though IDSes have been researched over 20 years, we still do not have an answer for the question of what features should be monitored.

So far, different kinds of methods and algorithms have been developed for anomaly detection but the focus has been on making them more efficient. Almost all of them are lacking the same information; what features are important for IDS, especially in telecommunications networks? For some reason information on the used features is not easily found from IDS research publications.

7.1.4.6 Next Steps

Previous sections are focused on information on the prior studies about data features and DDoS simulation environment.
As next steps, following tasks are to be completed.

- Choose appropriate detection method
- Decide proper measurement and metrics
- Experience detection method with DETER test bed and obtain initial results
- Develop feature extraction and selection algorithms

A data feature set will be constructed taking advantage of prior experiences on different datasets. These data features can be reduced or manipulated/changed after obtaining results from simulations.
Building a test bed itself will be a challenging work. Because of this reason DETER test bed will be used primarily for developing and testing DDoS detection algorithms. Using experiences gained from DETER test bed a new test bed suitable to our needs will be established.

7.2 Network activity simulation

7.2.1 Network simulation methods

In this chapter we will focus on the discrete event simulation only as it is widely used in software applications regarding computer network simulation.

The purpose of the discrete event simulation for computer networks is to experiment network behavior with models traducing the expected architecture, nodes characteristics, links and flows. A model is a simplified and observable representation of a real system, on which experiments can be conducted. Simulation aims at observing the system model behavior depending on time. The discrete-event principles are to describe sequences of events to be triggered, in some conditions, potentially from multiple sources of events at the same time, and then to analyze the overall system behavior due to changes in the system state. An event can be seen as the time when a resource state changes.

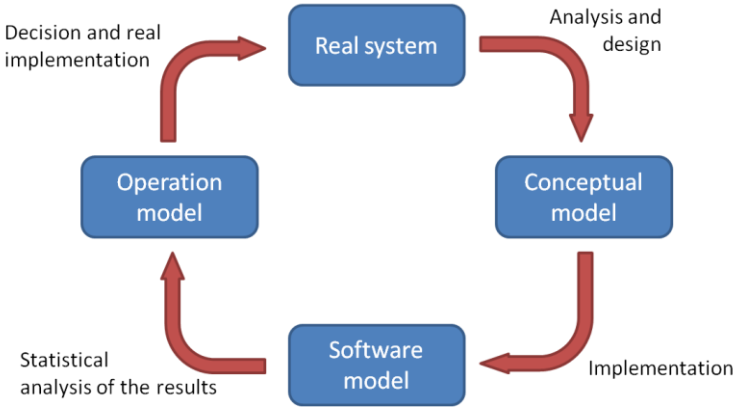


Figure 7-1: Development lifecycle

Getting a discrete event simulation necessitates to build the simulation model, which is a stochastic model, in the sense that components behavior may be described as not deterministic (e.g. component failure). What's more discrete-event models take the time dimension into consideration, i.e. these models are dynamic ones. This is achieved in several steps: analysis objectives definition, conceptual model definition (relevant elements and states, relationships, etc.), input model data specification, computer program implementation, computer program check and validation.

The basis for the discrete event simulation is to determine the scope of the real system to study, identify the different types of possible events occurring in the system (at least those in the scope of the study), and the associations between events and state changes (events that lead to a state, events resulting from state change). Then it is necessary to define an event scheduler describing a list of events and their occurrence time. Defining an event consists in explaining where it occurs on the system, which are the effects on the system and when the next event of the same type will occur. This step of event definition is a very huge and complex one but this is fundamental to get the most accurate simulation possible. It requires for instance statistical analyses of data used as inputs for simulation models, so that one knows how to reproduce realistic conditions. Once this is done, a software program has to be written from scratch or by filling a knowledge database of a dedicated simulation application (see examples of simulation systems in subsequent chapters).

A tricky step comes just after the implementation: the computational model validation that will give an idea of the level of approximation. Then simulations can be run, describing different situations.

7.2.2 Network simulation systems

7.2.2.1 NS-3

NS-3 is a discrete event simulator for Internet systems, targeted for networking research. NS-3 provides substantial support for simulation of TCP, routing, and multicast protocols over wired and wireless (local and satellite) networks. NS-3 is developed by ISI, the Information Sciences Institute at the USC School of engineering.

NS-3 is a set of modules playing different roles and/or targeting different purposes. Modules that express network behavior and elements are called models. NS-3 comes with a library of models that have been implemented by the NS team or NS community. For instance, the network module has been designed to model and manage network packets, nodes, sockets and queues. This is a module on which internet and mobility related modules rely on.

The creation of a simulation starts by the definition of the simulated network environment. First of all, topology elements and links have to be set. Typically, once has to define nodes (i.e. hosts for wired networks), net devices and channels and the way they are connected to. Topology helpers are available to simplify this stage. Then protocols used on these nodes have to be defined. Specific helpers exist too at this stage. For instance the InternetStackHelper allows for a quick installation of the TCP, UDP, and IP on a given list of nodes. IP addresses can be set and applications running on nodes too, more precisely the visible behavior in terms of network of these applications. This is done through events triggered at a given time.

Once every parameter has been set defining the environment, it is possible to run a simulation which is basically the execution of events scheduled during the preparation step. A logging system grants that simulation results remain available after the simulation, for further analysis.

NS-3 architecture allows for 2 kinds of simulation that can be combined with each other:

- Plug equipment (virtual or physical) to a simulated environment through an overlay of the network interface of the equipment. This piece of software is called a TapDevice. With this option, an existing device can participate to simulations, for instance to be tested under reproducible situations.
- Drive an equipment behaviour through a simulation and feed the real environment the equipment is connected to with data from the simulation. The link between the simulation and the driven machine is called EmuNetDevice.

NS-3 does not have a GUI to model and run simulations but NS-3 comes with two extensions dedicated to animations: NetAnim and PyViz.

NetAnim works with trace files generated by NS-3 during a simulation. It generates and displays the network nodes implied in the simulation and is also able to show animations regarding packet flow.

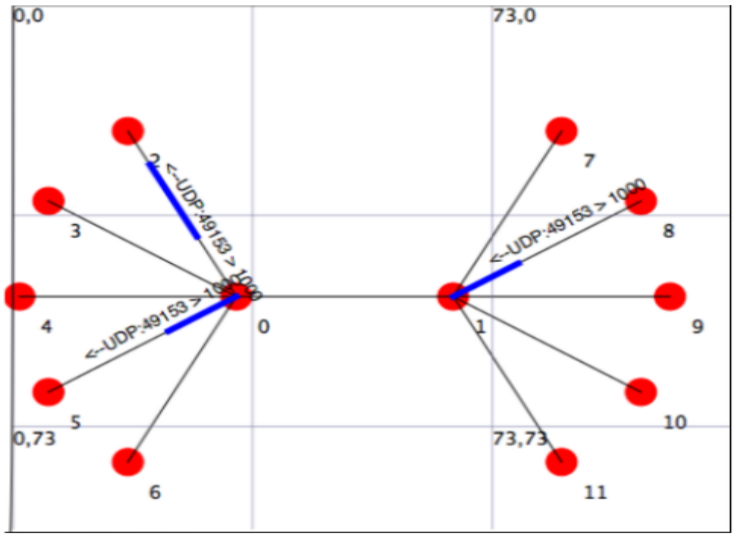


Figure 7-2: NetAnim screenshot (from NS-3 web site)

PyViz, a module written in Python, works live, in the sense that it does not need trace files. It is mostly adapted to debugging mobility models.

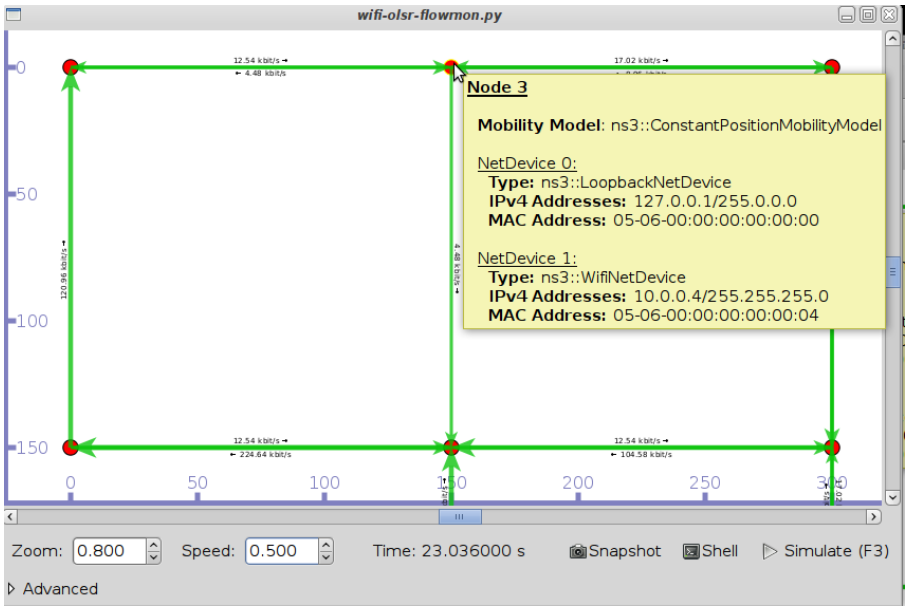


Figure 7-3: PyViz module screenshot (from NS-3 web site)

Availability:
The full source code of NS-3 can be downloaded and it can be compiled on multiple platforms, including most popular UNIX flavours and Windows.

Sources:

- NS-3 web site: http://www.nsnam.org/
- Wikipedia : http://en.wikipedia.org/wiki/Ns_%28simulator%29

7.2.2.2 OPNET

OPNET is a commercial software suite dedicated to application and network performance. OPNET has developed modules for network modeling and simulation.

OPNET's suite of products combine predictive modeling ad a comprehensive understanding of networking technologies to enable customers to design, deploy, and manage network infrastructure, network equipment, and networked applications. In particular OPNET Modeler is a development environment, allowing you to design and study communication networks, devices, protocols, and applications.

OPNET has been positioned in the leader's quadrant of the 2012 Gartner Magic Quadrant for application performance monitoring.

OPNET has interesting features for cyber security. OPNET offers a cyber-security suite which, probably, includes these four solutions: OPNET modeler, AppREsponse Xpert, OPNET nCompass and Sentinel, with pre-built packages dealing with cyber-attacks simulations.
These features are:

- Configuration Auditing: it provides an understanding of vulnerabilities in the network and shows what the impact of threats is. This solution is also able to evaluate automated vulnerability assessment (security compliance, port scan analysis), automated security posture tracking (network differences report), and network vulnerability assessment (impact of device failure).

Figure 7-4: OPNET Network configuration reports (from OPNeT web site)

- Modelling and Planning: it provides an evaluation of different network configurations and their resilience to cyber-attacks. It may also simulate cyber-attacks scenarios while using protocol and device library.

Figure 7-5: OPNET Network attacks and countermeasures (from OPNeT web site)

- Real-time Monitoring of Traffic and Applications: it provides the ability to collect information from network/applications, defines behavior or performance of them, monitor traffic, be alerted when something seems to be wrong or fail.

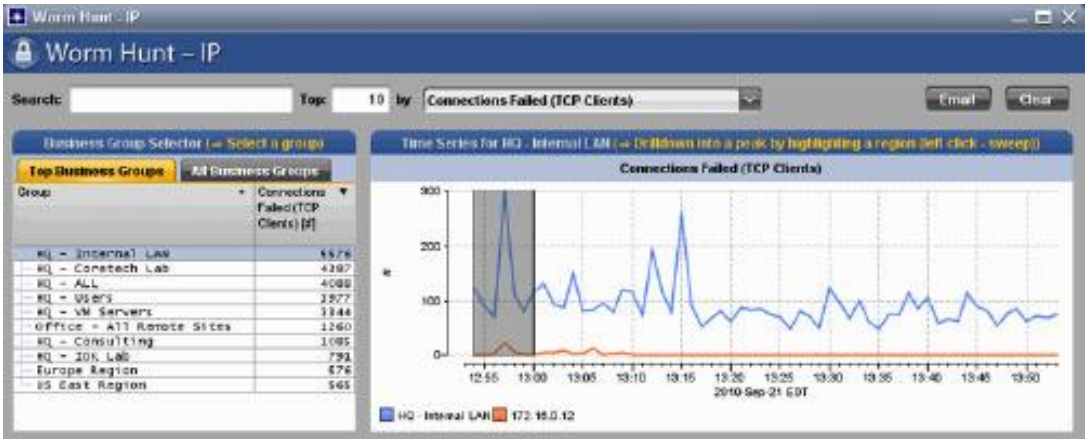


Figure 7-6: OPNET (from OPNeT web site)

- Training: it provides a simulation for cyber-attacks to be able to detect and mitigate them. That uses test bed to provide it. Thanks to that solution, you will be able to test real scenarios from military protocols and device models.

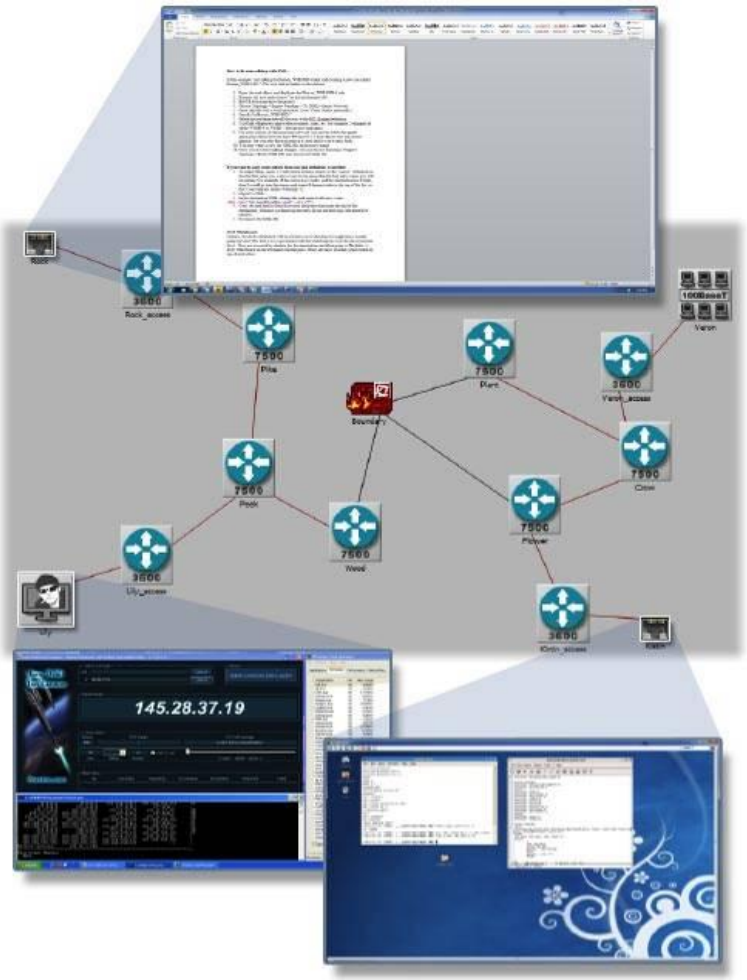


Figure 7-7: network attack and countermeasures (from OPNET web site)

This solution does not give a clear presentation of what kind of tool they are using to provide such capabilities. OPNET may want to enlarge their own tools by including cyber security aspects but this is not yet well included and defined.

As a conclusion, the biggest advantage of OPNET is to propose an exhaustive library of telecommunication protocols, mainly on packet switching. Furthermore OPNET is very open et allows an expert user to develop his/her own models. But OPNET has some drawbacks and usage is very complex. This is the consequence of its exhaustiveness. The most noticeable drawback is about slow response time when it comes to simulating real networks.

Availability:

- Commercial license

Sources:

- OPNET web site: http://www.opnet.com/solutions/network_management/cyber-security

7.2.2.3 NetSim

NetSim™

NetSim is developed by Tetcos, an Indian company, and it is used for network laboratory experimentation and research.

NetSim is a stochastic discrete event simulator that comes in two versions: the academic version and the standard version.

The academic version provides a traffic generator using models that reproduce data or voice transmission. Additional tools giving performance metrics and captured packet traces.

The standard version has the same features as the academic version, plus features protocol source codes in C which can be modified and linked via the development environment. A development environment and a simulation reporting module are embedded in this version.

Whatever the version, the protocols covered in simulation are aloha, slotted aloha, Ethernet - CSMA / CD, Fast Ethernet, Gigabit Ethernet, Token Ring, Token Bus, W-Lan, X.25 Frame Relay, ATM, TCP, IP -Routing RIP, OSPF, BGP, GSM, MANET, MPLS, Wi-Max, Wireless Sensor Networks and Zigbee 802.15.4.

As this is a tool for exercises, a set of default exercises is available with the product.

Availability:

- Downloadable from editor web site, an activation key needed for each node the software simulator is running on.

Sources:

- NetSim (Tetcos) web site http://www.tetcos.com/index.html
- Wikipedia: http://en.wikipedia.org/wiki/NetSim

7.2.2.4 GloMoSim

GloMoSim
Global Mobile Information Systems Simulation Library

GloMoSim stands for Global Mobile Information Systems Simulation Library. It is a scalable simulation environment for wireless and wired network systems. It employs the parallel discrete-event simulation capability provided by Parsec that is a C-based simulation language, developed by the Parallel Computing Laboratory at UCLA, for sequential and parallel execution of discrete-event simulation models.

Several network nodes described in the simulation environment are a single Parsec entity to avoid performance degradation at initialization stage. Such an entity represents a geographical area. That means network nodes are gathered depending on their physical location. This is called network gridding in GloMoSim. It is important to note that the state of each node has been expressed independently in order to get relevant results.

GloMoSim follows a layer approach that fits to the network layer model, with layers aggregated into a single Parsec entity.

Supported protocols are: Random waypoint, Random drunken, Trace based, Two ray and Free space, Noise Accumulating, SNR bounded, BER based with BPSK/QPSK modulation, CSMA, IEEE 802.11 and MACA, IP with AODV, Bellman-Ford, DSR, Fisheye, LAR scheme 1, ODMRP, WRP, TCP and UDP, CBR, FTP, HTTP and Telnet.

Availability:

- GloMoSim source and binary code can be downloaded only by academic institutions for research purposed. Commercial users must use QualNet, the commercial version of GloMoSim.

Sources:

- GloMoSim web site: http://pcl.cs.ucla.edu/projects/glomosim/
- Parsec web site: http://pcl.cs.ucla.edu/projects/parsec/

7.2.2.5 QualNet

QualNet is a commercial product developed by SCALABLE Network Technologies (SNT). It is a communications simulation platform designed for testing, planning and training.

Qualnet has different modules to set up and run simulations: Qualnet Architect, Qualnet Analyzer, Qualnet Packet Tracer, Qualnet File Editor and Qualnet Command Line Interface.
Through these modules, it is possible to define new protocols models, to update existing ones, to design networks and to analyze the simulation results.

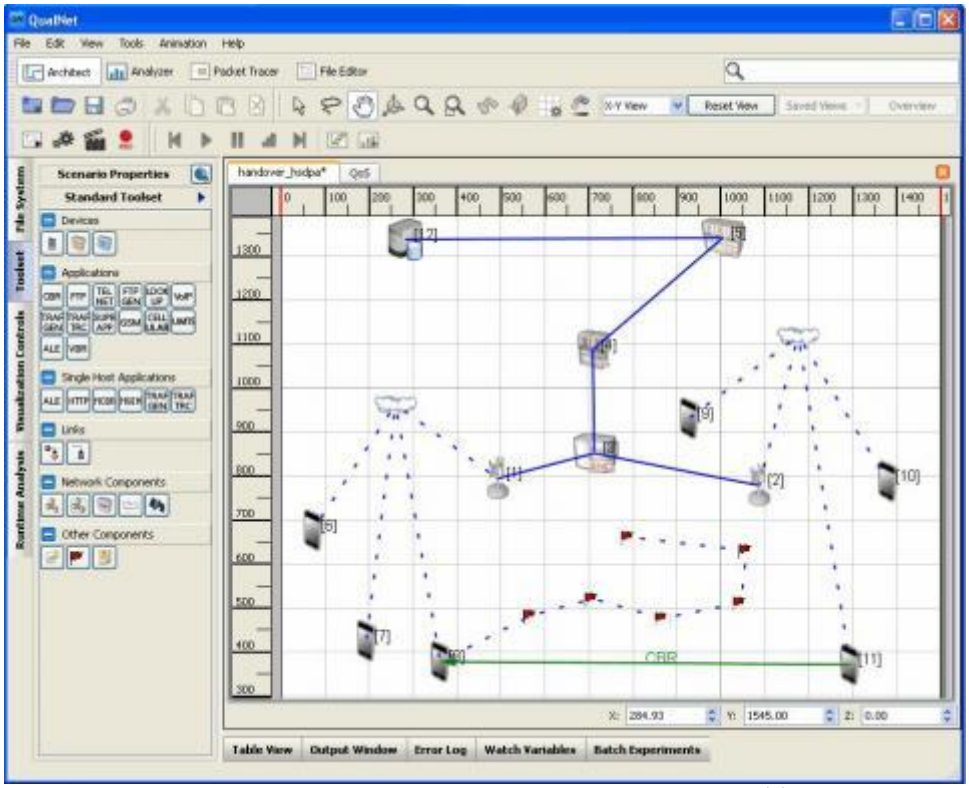


Figure 7-8: Qualnet screenshot, scenario builder (from Scalable Network Technologies web site)

Availability:

- Commercial license

Sources:

- QualNet (SCALABLE Network Technologies) web site: http://web.scalable-networks.com/

7.2.2.6 EXata/Cyber

EXata/Cyber is a commercial product also developed by SCALABLE Network Technologies (SNT). EXata is the simulation and emulation platform. The Cyber testing environment is made up of the Cyber Dynamics Library from SNT. It aims at verifying the resiliency of communications such as modeled in EXata towards cyber-attacks.

Cyber Dynamics Library includes a large range of different possible attacks. Thanks to that repository, this solutions is able to simulate cyber-attacks and specifies what kind of attack you want (DDoS, eavesdropping, etc.) and which target is targeted (Wireless, wired, mobile ad-hoc, etc.). This solution also includes several cyber dynamics models like IPsec, WEP/CCMP, Certificate Model, etc.

The picture below shows interactions between wireless device and their range.

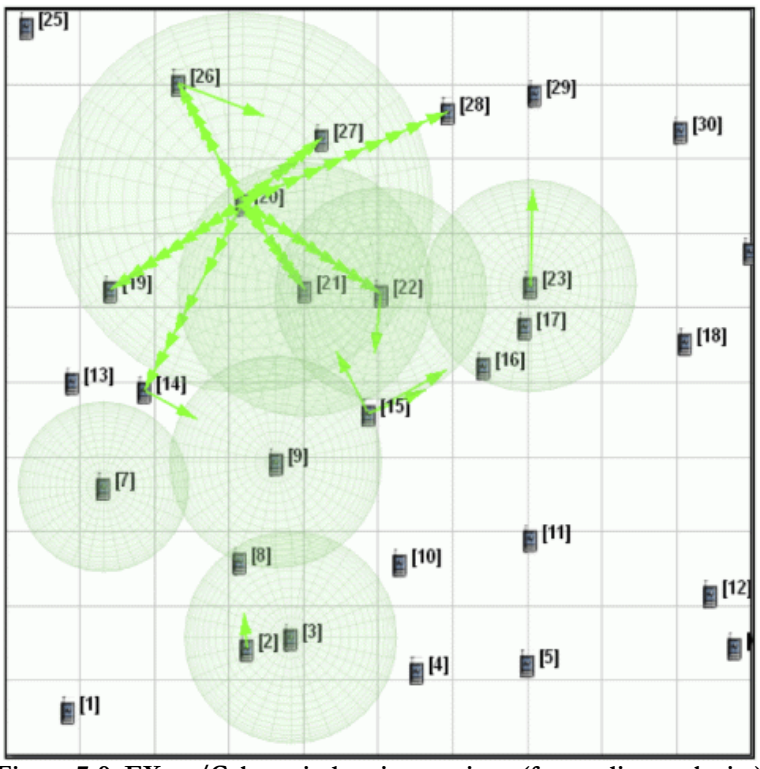


Figure 7-9: EXata/Cyber wireless interactions (from editor web site)

The picture below displays links which are currently operated between different devices and information being transmitted.

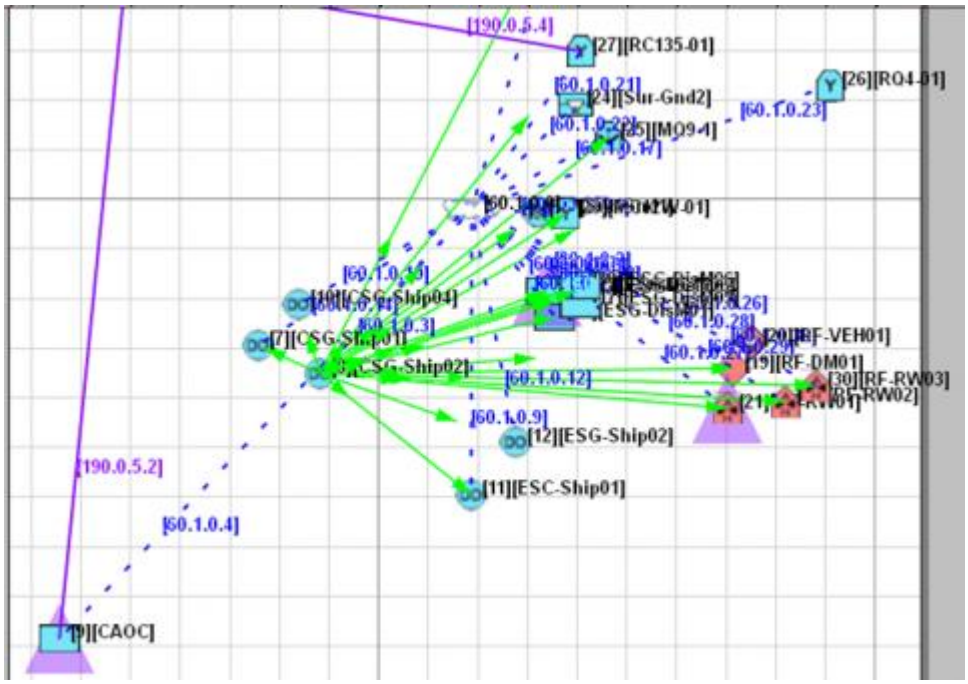


Figure 7-10: highlighted transmissions (from editor web site)

Moreover, EXata/Cyber also includes the EXata platform. This bundle provides an emulator to create and evaluate communication network. EXata can use an existent network or create it. Thanks to that solution, users will be able to develop their own network topology, design new network, implement real network, view real network with emulated network.

The picture below demonstrates the capability of EXata to emulate network

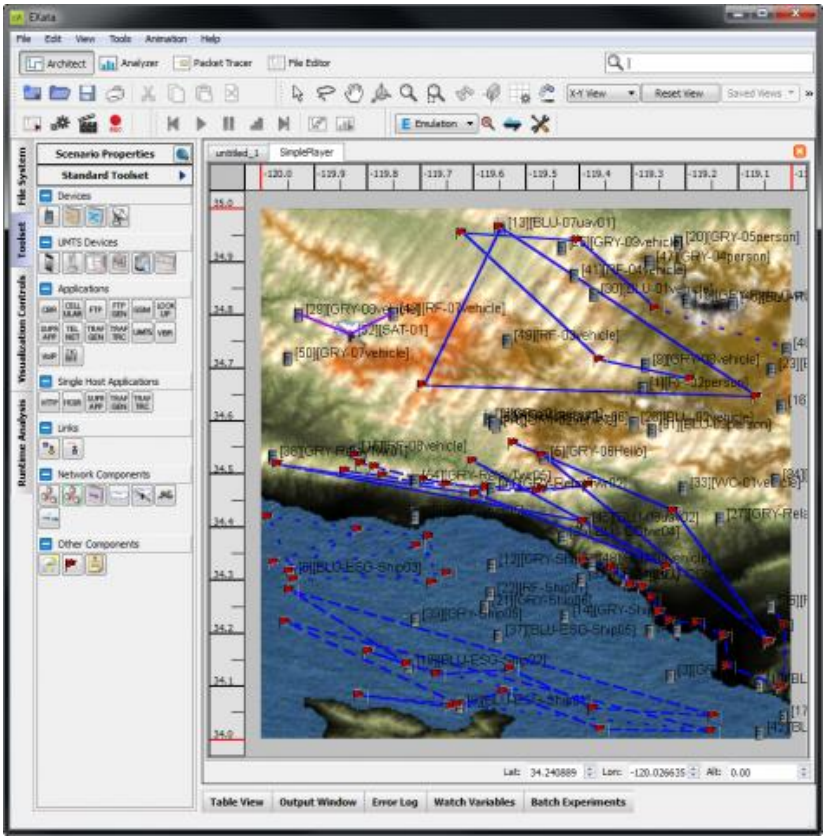


Figure 7-11: EXata/Cyber Network emulation (from editor web site)

The picture below provides a real 3D representation of network that is more suitable to understand for common people.

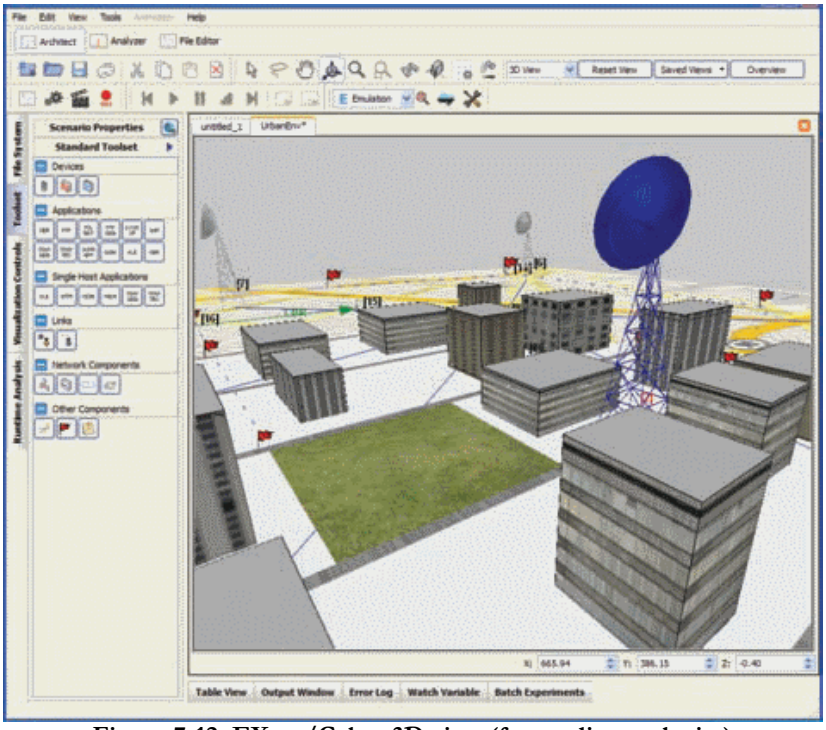
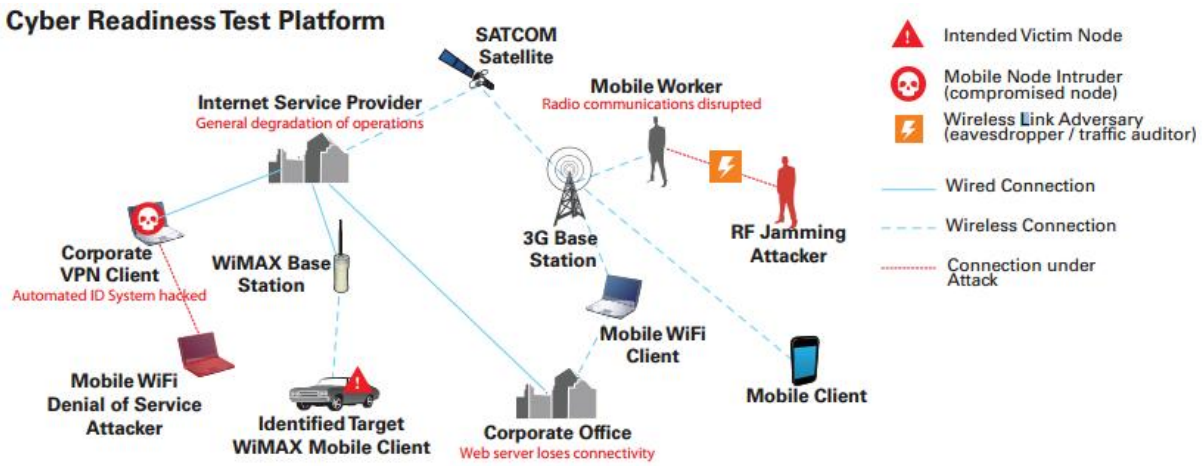


Figure 7-12: EXata/Cyber 3D view (from editor web site)

EXata/Cyber creates Software Virtual Network (SVN) that makes it possible to represent the communications infrastructure at such high fidelity that applications running on it. This is one of the possibilities this solution can reach to:



This solution clearly defines what kind of tools are used, offers a 30-days evaluation software and supplies model libraries to express elements (such as cyber, equipment, interfaces, protocols, etc.).

Availability:

- Commercial license

Sources:

- EXata/Cyber (SCALABLE Network Technologies) web site: http://web.scalable-networks.com/content/exatacyber

7.2.2.7 OMNeT++

OMNeT++

OMNeT++ is a high-performance object-oriented discrete event simulation environment for building network simulators. Targeted networks are wired and wireless communication networks.
The simulator can be used for modeling: communication protocols, computer networks and traffic modeling, multi-processors and distributed systems, etc. OMNeT++ also supports animation and interactive execution.

It is a component-based, modular and open-architecture environment with strong GUI support and an embeddable simulation kernel (in charge of event scheduling, result recording, etc.), as a C++ library. Models are written in C++, respecting the OMNet++ simulation architecture. Describing a full simulator is possible through the assembly of modules, each of them being a particular model. The module communications are done in a message-passing mode (indirectly by requests, or directly sending information). OMNet++ comes also with a publish-subscribe mechanism that lets publisher modules emit messages that will be propagated to the subscriber modules.

A recording functionality allows for recording at the framework level (using notification signals) or directly in the module. OMNet++ gets a package dedicated to read the recorded results of a simulation

The development environment is based on Eclipse C/C++ Development Tooling, CDT (cf. Eclipse web site for more details, http://www.eclipse.org/eclipse/index.php), as this framework is specifically adapted to object-oriented development. In this environment, some graphical interfaces are available for simulation execution and analysis.

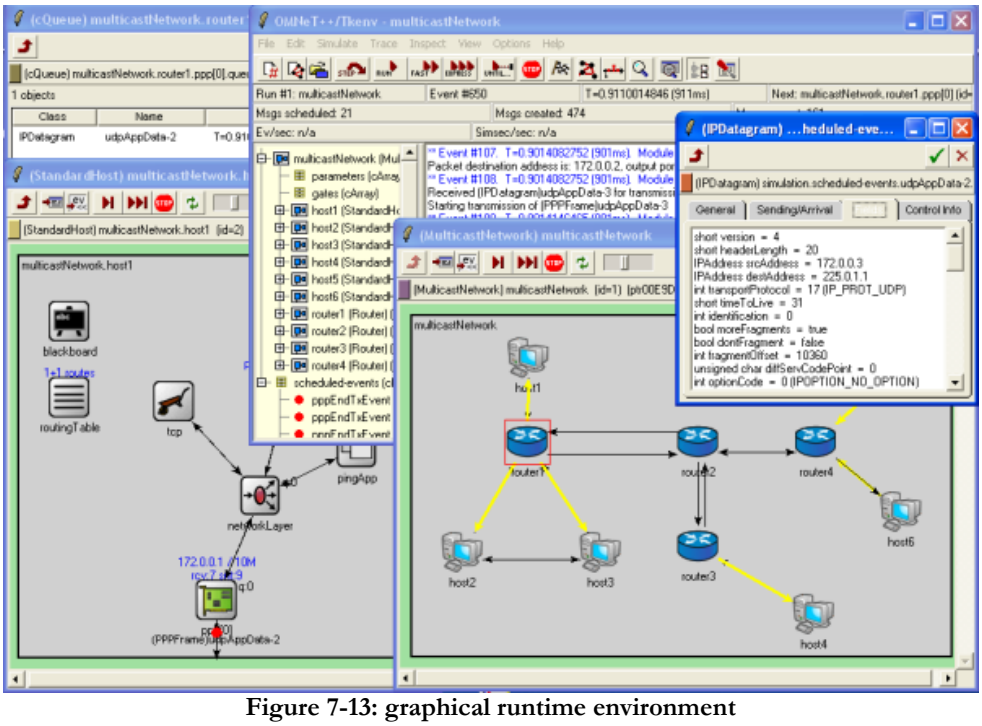


Figure 7-13: graphical runtime environment

The NEtwork Description (NED) tool is a graphical editor that helps in defining network components and modules parameters (traffic sources, queues, delays, jobs, job forks and joins, parallelism of processes, etc.). There is also an editor for configuration files. Note that there are pre-built blocks that help in defining network components.

There is a commercial version of OMNeT++ called OMNEST. In the commercial version, packaging and licensing mode are different (windows installer in the commercial version, commercial license vs. academic license) and some features, such as SVG image export, are present in the commercial version. A support service is provided in the commercial version.

Availability:

- OMNeT++ is freely distributed under an academic public license and can be downloaded from http://www.omnetpp.org/omnetpp. OMNeT++ is free for academic and non-profit use, and it is a widely used platform in the global scientific community. Commercial users must obtain a license from omnest.com
- Even in the commercial version, the full source code is provided.

Sources:

- OMNeT++ web site: http://omnetpp.org/
- OMNEST web site: http://www.omnest.com/
- Wikipedia: http://en.wikipedia.org/wiki/OMNeT%2B%2B

7.3 Security impact simulation

7.3.1 Systems

7.3.1.1 Skybox Security

Skybox Security provides five different solutions; Firewall Assurance, Network Assurance, Risk Control (all of these 3 solutions are gathered into the Skybox View Enterprise Suite), Change Manager and Threat Manager. Each of these solutions provides ticketing and reporting system. All figures that are presented below are snapshots from demos of the Skybox Security software.

Firewall Assurance

- It gives to IT teams firewall management power to find and remediate firewall security issues, optimize firewall rule sets, reduce configurations errors and test planned firewall changes in advance. First of all, this solution collects firewall configuration data, log files and corporate security policies. A normalized firewall configuration repository is created, allowing evaluation of different firewalls. Configurations may be entered manually or collected automatically from existing configuration repositories. Then, it analyses and correlates firewall configuration and policy data with industry standards and firewall management best practices. Security and compliance gaps are identified and prioritized to show firewall administrators the areas of greatest concern. Globally, Firewall Assurance provides 4 main functions

  - **Policy compliance**
    This function contains 2 types of compliance; the access policy looking at the traffic flow through a device and the rule policy looking about the syntactic checking of the rule, e.g. if destination or services are set to “any”.

➢

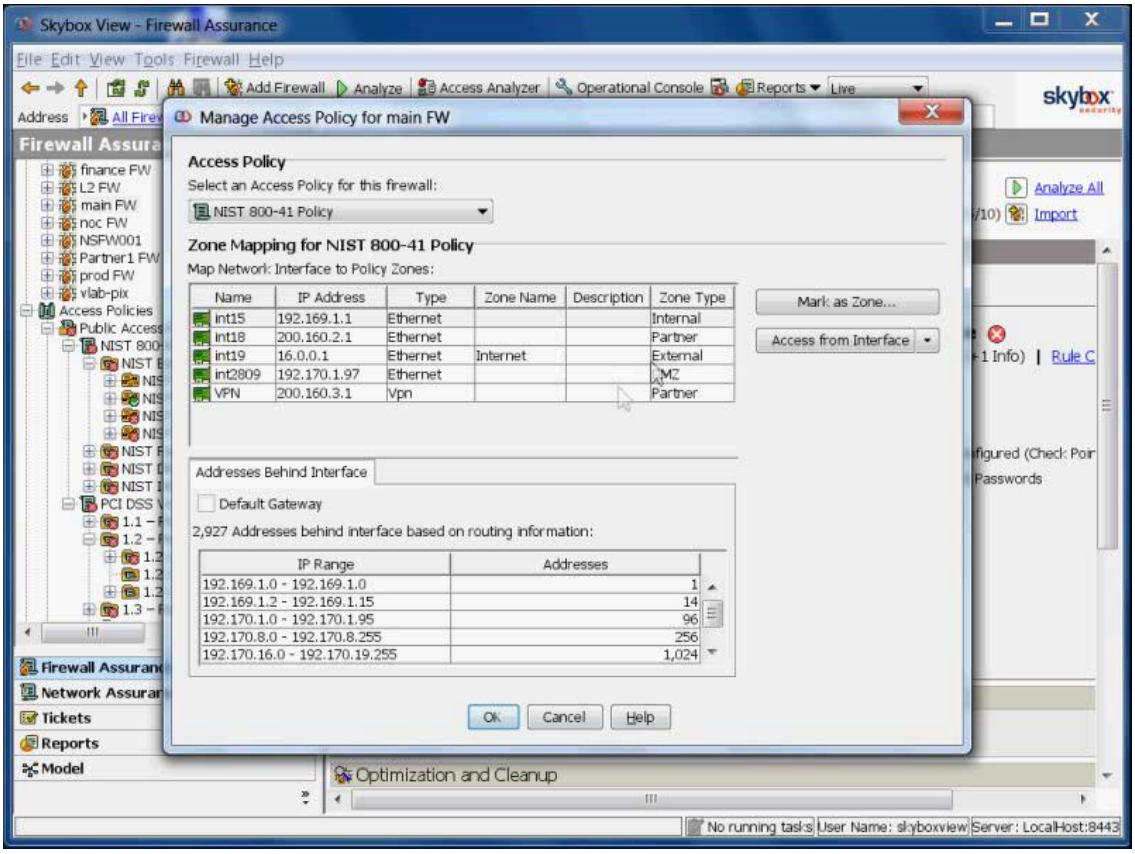


Figure 7-14 : Access policy

By default, NIST 800-41 and PCI DSS v1.2 are installed. But it is possible to customize specific policy or add other compliance standard. For each of firewall, a percentage is present to show how it respects the standard into the network. So if there is any change in firewall configuration, the percentage will be modified and there will be an impact on the policy requirement. It is also possible to define if there is an exception.

- **Configuration compliance**
  Some standard configuration policy is available by default (such as Cisco IOS, Netscreen). This section provides a tool which analyses the current firewall configuration (e.g. default admin username used).

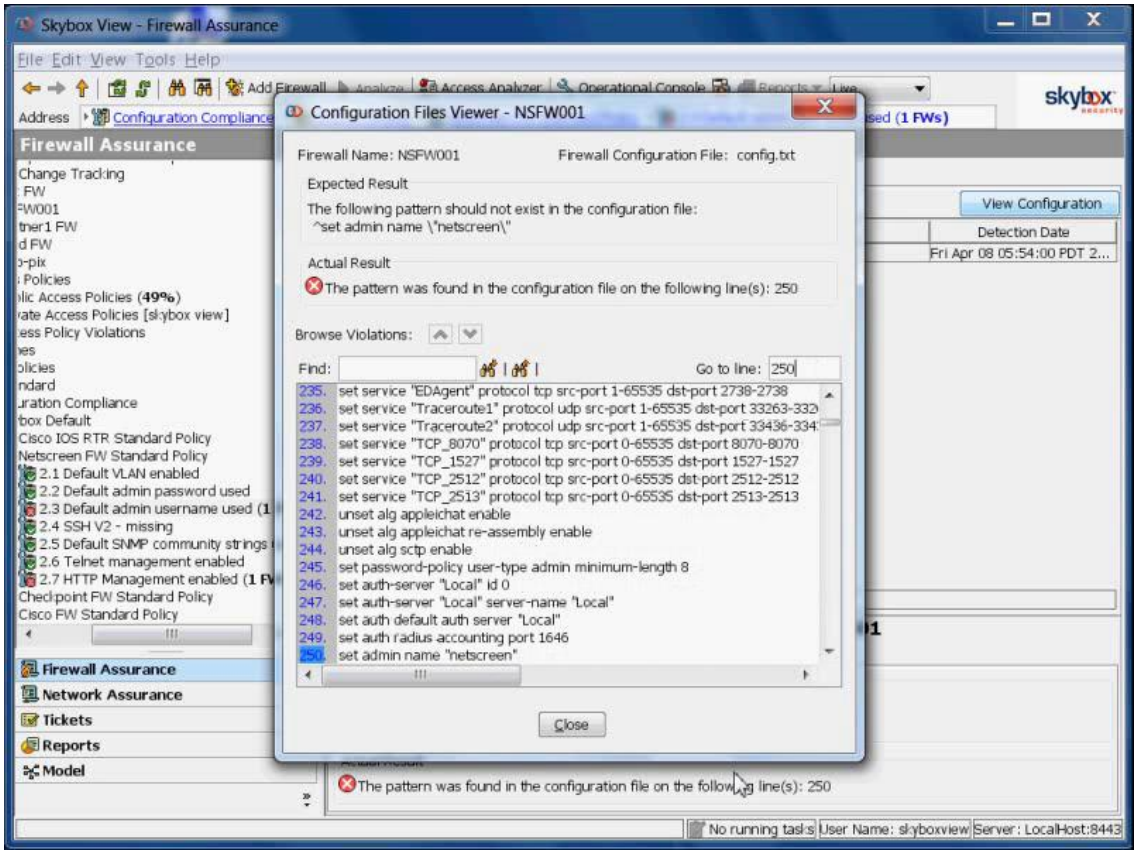


Figure 7-15 : Configuration compliance

- **Optimization & Clean-up**
  This function contains 2 parts. Shadowed and redundant rules (this allows to find rules that cannot be used because they are covered by other rules), and Rule usage (look at the log of the firewall on seeing which rules are hit and how many hit they have to know which rules are used and unused).

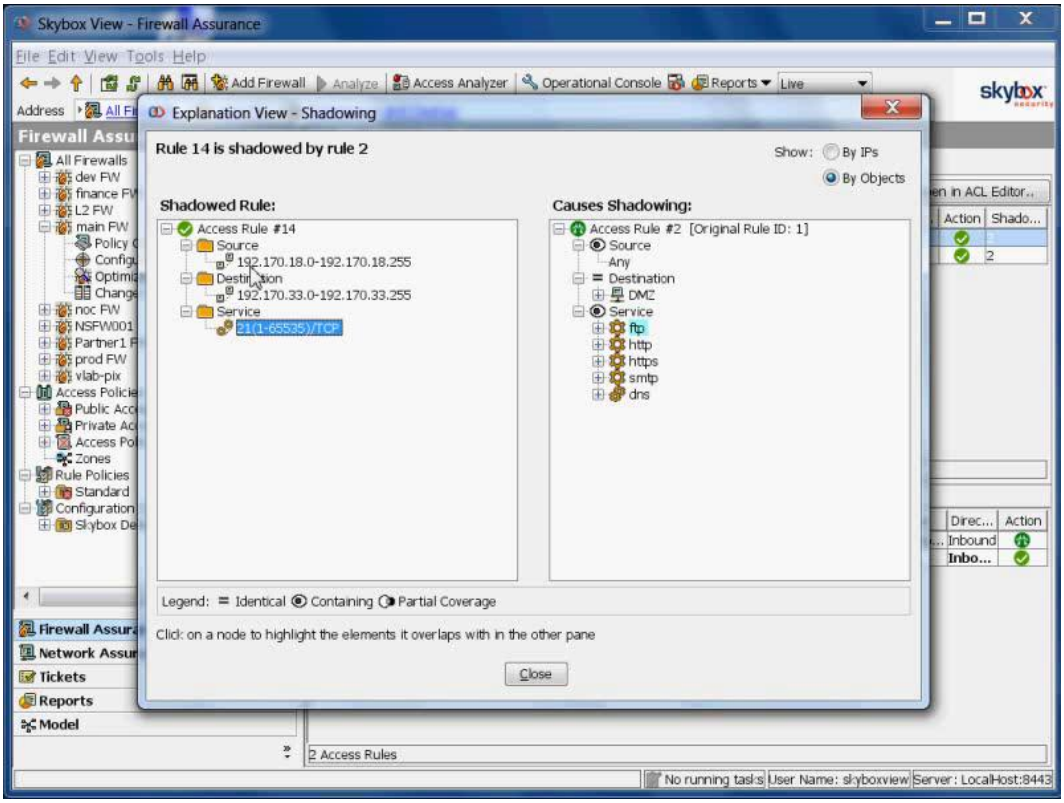


Figure 7-16 : Shadowed rule

- **Change tracking**
  This section provides an historical log of all the changes that have occurred in a firewall.

- **Network Assurance**: it provides comprehensive visibility of the network topology required to ensure security and compliance, and enables fast analysis of difficult network configuration and connectivity questions. It automatically collects and analyses network configuration data and provides in-depth visibility with a detailed network model and topology map. It enables operators to troubleshoot network device configurations, analyse access paths and preview changes on the network model, without impacting the live network. It provides complete access path analysis, including all aspects of the network infrastructure such as routers, switches, proxies, all ACLs, routing rules. It analyses network impact of security and availability requirements before change is made. First of all, it collects configurations data from all network devices. Then it maps and analyses data configuration to create a model and a visual map of the network. Network changes and availability issues are evaluated quickly with no impact to the live network. It also provides a map visualization to see network in more common way. To create a model of the network, Skybox imports all layer 3 devices accomplished by repository or connected all devices and send their information by protocol such as SSH. But there is no information about if it is possible to add a new device on the model. An important information is that, Skybox produces a virtual model of the actual network, it calculates all the dynamic routes that can be reached for each device and service, and this is not online. Any modifications that are made here will not affect the actual network.

- **Access analyser**
  This function allows seeing specific query. For example, it is possible to identify which network and which services are reachable by internet. This also allows making changes to see the impact and what devices should be modify in order to gain access (e.g.: allow internet access between 2 partners, the tool will provide a list of which devices have to be modified to allow this access).

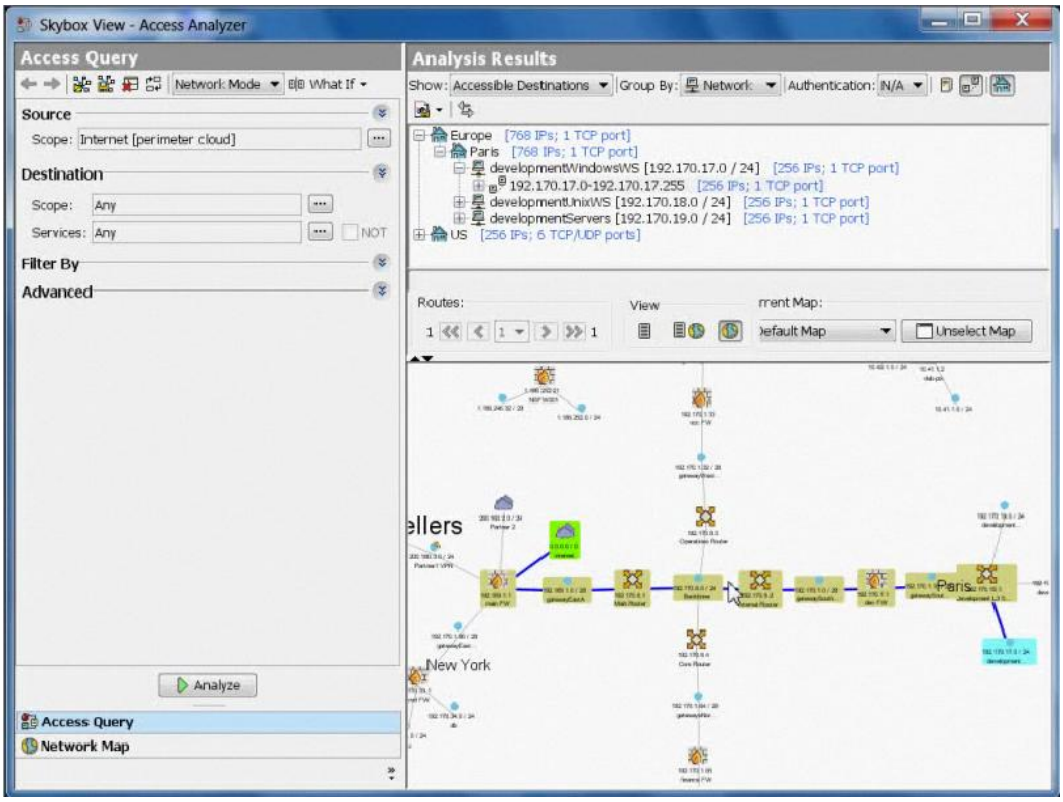


Figure 7-17 : Access analyser

- **Risk Control**: this security risk management solution combines capabilities to discover vulnerabilities daily, automatically evaluates risks and drives remediation activities. It automatically collects vulnerability, asset and configuration data from network devices and management systems. It runs attack simulation daily to identify exploitable vulnerabilities. It evaluates remediation alternatives and provides decision support and tracks progress of remediation activities. First of all, it deduces a list of vulnerabilities with Vulnerability Detector and collects data from threat feeds and vulnerability scanners. Then it analyses the collected information to create a model of the network and incorporates Skybox Vulnerability Content (using its own vulnerability dictionary, CVE dictionary and CVSS scoring method) with the likelihood and severity of potential attacks. Consequently, this Skybox module can be used for prevention. It looks interesting as a complement to the core targets of the project, which are detecting on-going complex attacks and assessing the security impact of the potential attack countermeasure.

- **Threat Manager**: it provides an improvement of Risk Control tool to easily manage threat workflow by continuously presenting the latest update of an alert, highlighting matches in the system and executing queries that automatically prioritize vital threats. It identifies the most critical threats to the organization, by calculating their CVSS score. It streamlines remediation process by automating ticket creation. First of all, it collects data from threat feeds and bulletins from Skybox Vulnerability Content. Then it analyses data to prioritize threats based on relevance and criticality to the organization. Remediation recommendations are generated and tickets can be issued automatically.

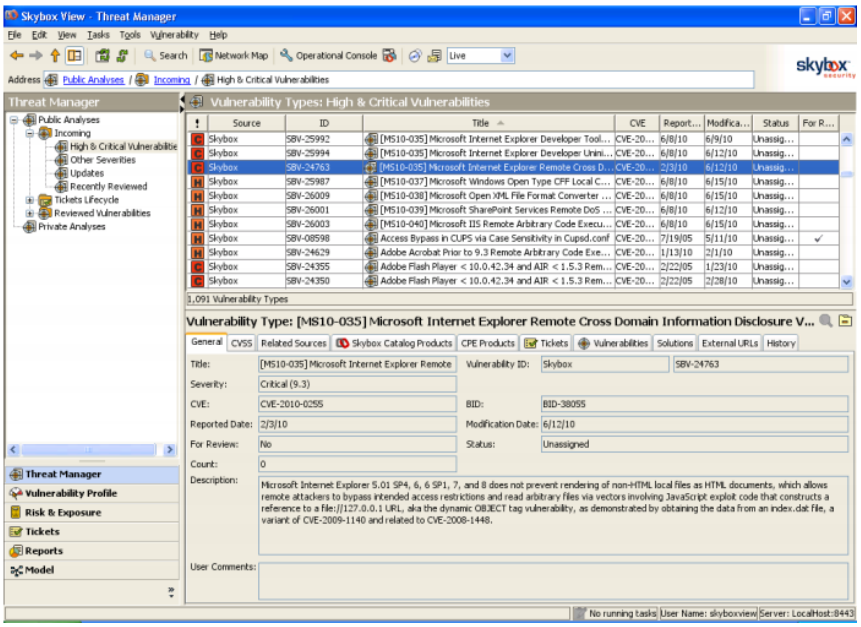


Figure 7-18 : Skybox threat manager

- **Change manager**: it adds complete change workflow capabilities to Skybox Firewall Assurance. It allows administrators to automatically capture, assess, plan and verify all firewall change requests. It identifies all relevant devices impacted by the change and assesses the request for security risks, vulnerabilities and compliance issues. First of all, it captures information from a short description of the change request. Then it finds the relevant firewalls and checks whether the connectivity is already permitted or not. It analyses the change request and validates that changes are in compliance with network policies and also identifies any violations that will occur if the change is implemented as well as the associated risk level. That will also provide a list of vulnerabilities exposures if the change is made. There is not a lot of information about this new solution and the major feature that can be useful for the ADAX project is the list of vulnerability exposures that can occur if any changes are made.

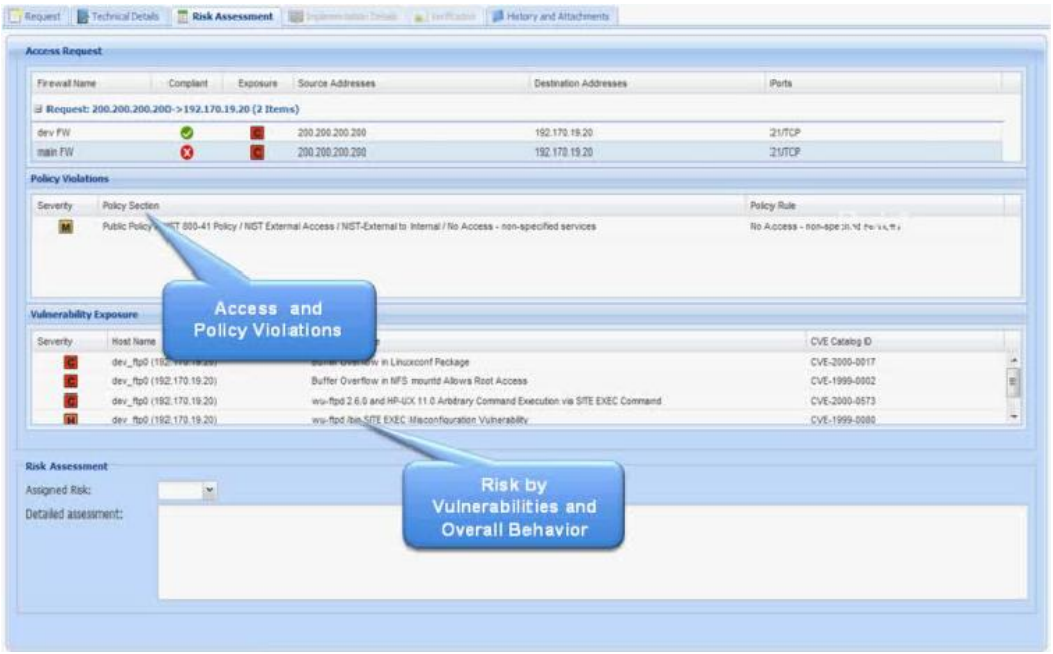


Figure 7-19 : Risk Assessment/Impact Analysis

Skybox Security provides a powerful set of tools. But for this project, the Network Assurance seems to fit the project needs in terms of simulation more than the others. Indeed, it provides a tool that can plan changes without impacting the live network. Unfortunately, it is not probably the main function of this tool and there is not a lot of publicly available information about it. Firewall Assurance also provides that kind of capabilities but is restricted to firewall configuration changes.

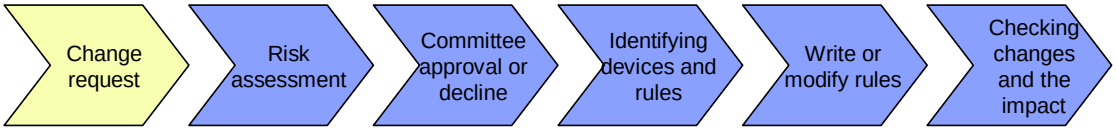
7.3.1.2 RedSeal

RedSeal Networks develop, through their solution RedSeal Platform 6.5, proactive security intelligence software that organizations depend on to visualize their security effectiveness, maintain continuous compliance with regulations and protect their most critical assets and data. Unlike systems that measure the impact of attacks after they transpire or address individual elements of networks protection, RedSeal analyses the cumulative ability of defences to control access and mitigate vulnerability exposure across the entire enterprise, providing the critical metrics necessary to trend performance and isolate gaps before they can be discovered by hackers.

RedSeal provides a solution to evaluate and measure the risk inherent to any modifications that organizations have planned. This may be adapted to the ADAX objectives regarding the evaluation of a countermeasure impact before enforcement of this countermeasure.

Note: Gathered information mainly comes from a white paper provided by RedSeal. Sections hereunder will describe how RedSeal Platform manages the risk of network changes.

This solution is divided into 6 parts. All features presented in this document are available through RedSeal Platform 6.5



Step 1: initial change request

An initial change request can be made from a wide variety of sources. For most of them those changes are driven by business objectives such as new service or new connection with partner. So, the main objective is to gain quick and secure access. A key to being prepared for change request is maintaining detailed knowledge about every possible point of access throughout the network infrastructure. To solve that issue, RedSeal offers an automatic and continuous models access across layer 3 (network layer in OSI model) of the entire network. Therefore, change requests can be evaluated.

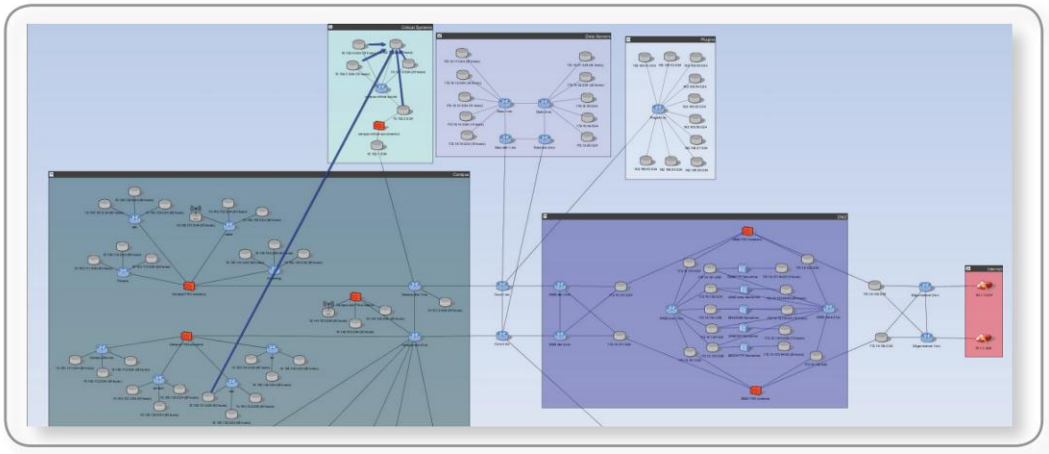
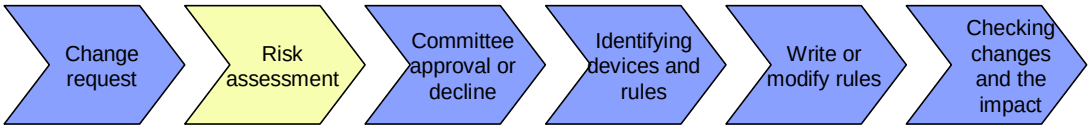


Figure 7-20 : Continuously updated network access maps ensure change management decisions are based on today's network



Step 2: Risk Assessment

When a change is requested, RedSeal automatically provides the change management committee with an automated risk assessment. This facilitates informed decisions about approving, conditionally approving or denying requests.

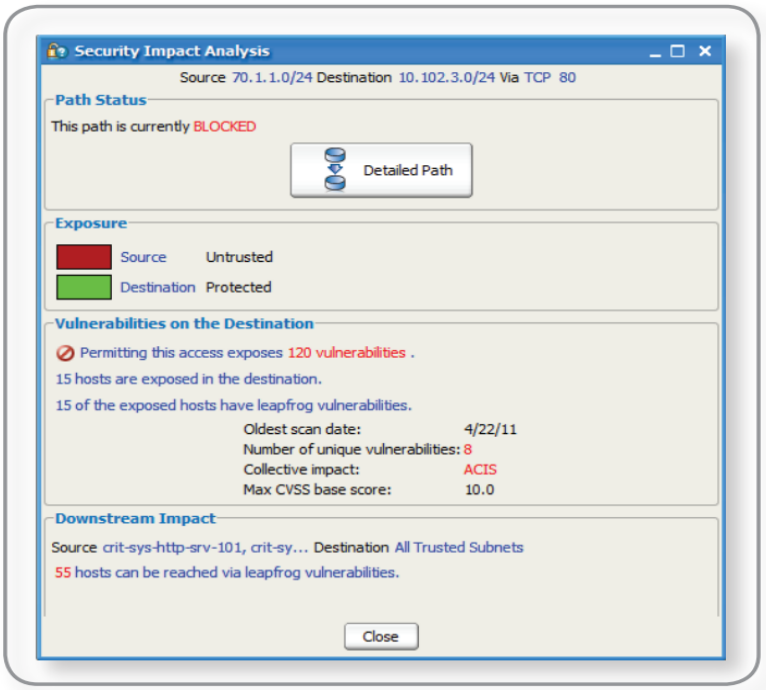
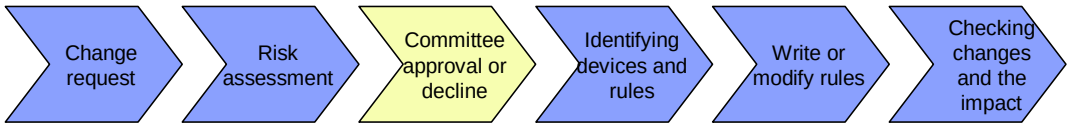


Figure 7-21 : Risk Assessment of change requests shows current status and potential risk or approving



Step 3: Committee approval or decline

The change committee must ensure that the correct level of priority is given to the change request and not creating redundant access rights already in place. Pressure from business operations to approve change requests quickly can strongly influence the approval process. RedSeal helps overcome objections and alleviate this pressure by presenting clearly defined risk criteria. RedSeal offers four different decisions to the risk assessment; request unnecessary (access already in place), approved (access not in place but no risk introduced if enabled), conditional approved (need patch or modify few thing before approval), denied (access exposes too much risk and requires too much modifications/patches). RedSeal also provides a centralized system of record for change management committee decisions. Moreover, in this section, one can enable approvals at the policy level (e.g. Internet access is allowed for mail servers on port 25) and track date, justification and employee providing the approval.

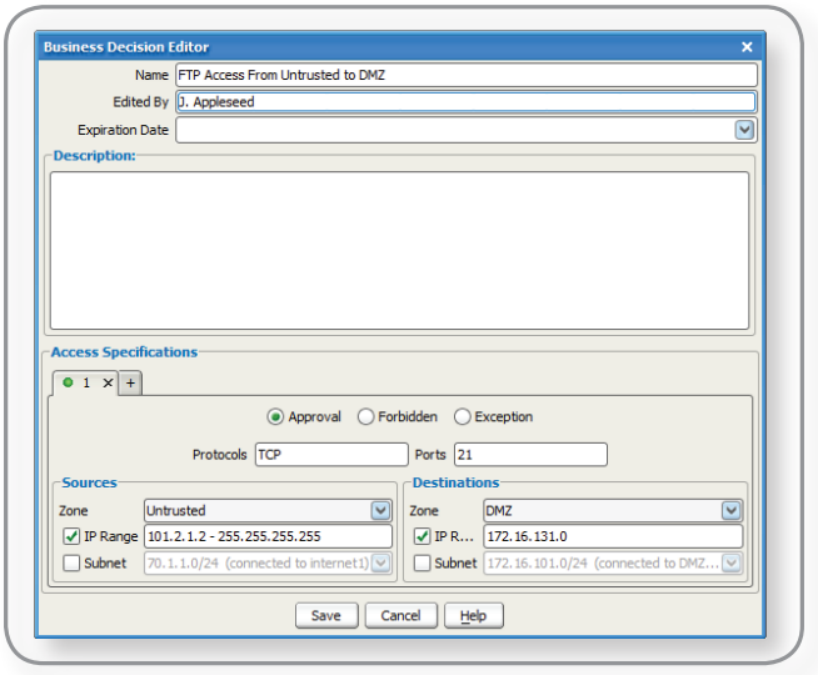
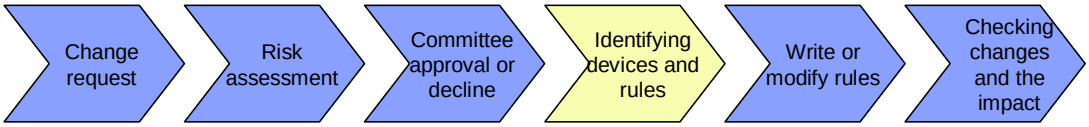


Figure 7-22 : Change approval tracking



Step 4: Identifying devices and rules

Accepted changes require a detailed analysis that involves translating the business request into physical changes within the network. To implement a change request, engineers must trace and track every possible path of network traffic to determine which devices will have an effect on data. That can take several days to accomplish this task. RedSeal solves this issue by analysing network infrastructure to automatically identify the devices and rules that must be changed. It provides a current map of the network infrastructure identifies all devices involved and isolates all rules and ACLs (Access Control Lists) that currently block the requested asset.

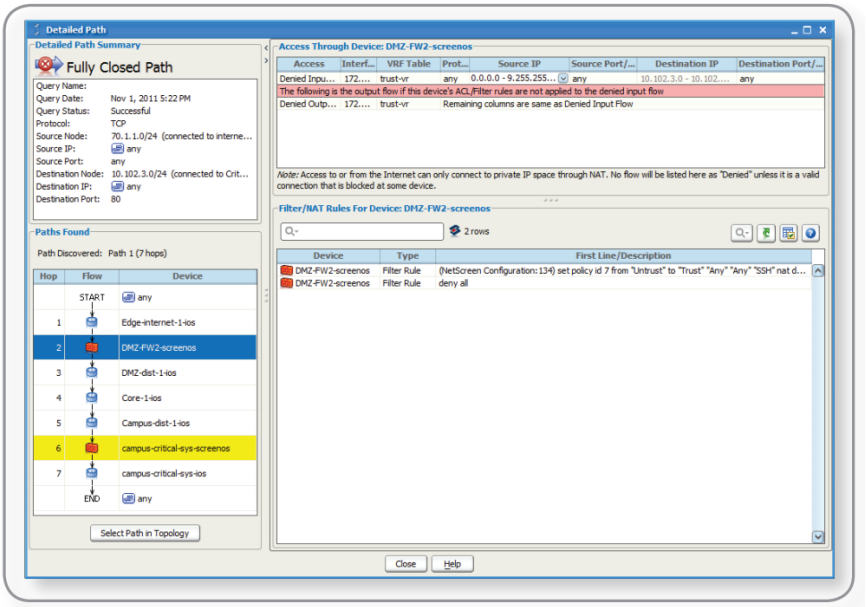
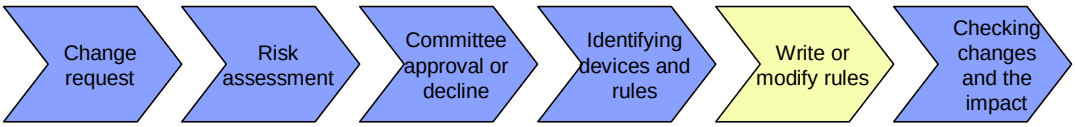
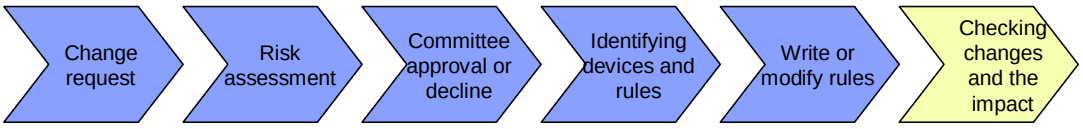


Figure 7-23 : Automatically identify the network path of the requested access and the filter rules that currently control access



Step 5: Write or modify rules

Nowadays, change management call for new rules to be written to enable each approved change. This creates a build-up of overlapping and redundant rules in network devices. RedSeal identifies the devices and rules that require modification and shows a clear presentation of it to the implementation team.



Step 6: Checking the changes and the impact

Whenever a change occurs, there is a possibility for errors. Even if network security does not change, it is possible that changes in the rest of the IT infrastructure may inadvertently create a security breach. RedSeal reviews and analyses the collective effect of all ACLs and rules in the enterprise and ensures that all resulting accesses have received policy level approval.

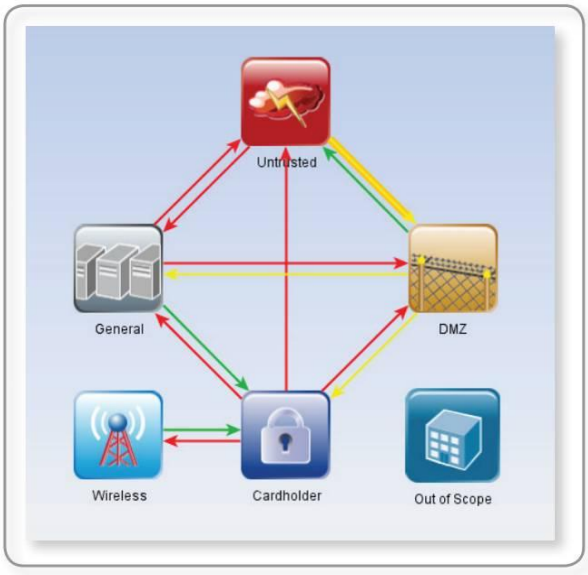
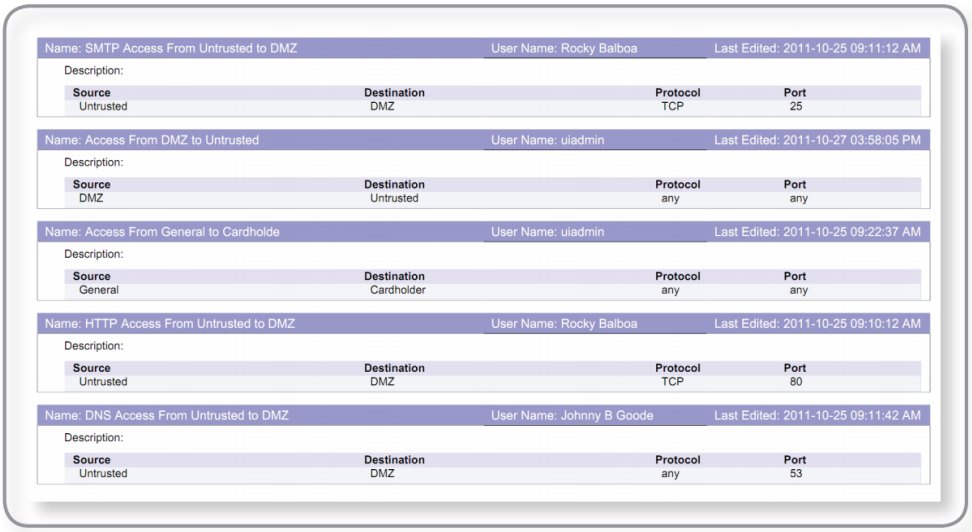


Figure 7-24 : Continuously monitor all security policies to ensure any changes don't result in policy breaches

If a breach is detected due to the change, RedSeal issues an alert. The operator can see which device has caused the breach. Moreover, many regulatory regimens, such as PCI DSS (Payment Card Industry Data Security Standard), require organization to demonstrate that all changes to critical data access have been approved. RedSeal provides detailed reports of the approved access and its justification.



Name: SMTP Access From Untrusted to DMZ		User Name: Rocky Balboa	Last Edited: 2011-10-25 09:11:12 AM
Description:			
Source	**Destination**	**Protocol**	**Port**
Untrusted	DMZ	TCP	25
Name: Access From DMZ to Untrusted		User Name: uiadmin	Last Edited: 2011-10-27 03:58:05 PM
Description:			
Source	**Destination**	**Protocol**	**Port**
DMZ	Untrusted	any	any
Name: Access From General to Cardholde		User Name: uiadmin	Last Edited: 2011-10-25 09:22:37 AM
Description:			
Source	**Destination**	**Protocol**	**Port**
General	Cardholder	any	any
Name: HTTP Access From Untrusted to DMZ		User Name: Rocky Balboa	Last Edited: 2011-10-25 09:10:12 AM
Description:			
Source	**Destination**	**Protocol**	**Port**
Untrusted	DMZ	TCP	80
Name: DNS Access From Untrusted to DMZ		User Name: Johnny B Goode	Last Edited: 2011-10-25 09:11:42 AM
Description:			
Source	**Destination**	**Protocol**	**Port**
Untrusted	DMZ	any	53

Figure 7-25 : Detailed reporting of changes and approvals

Figures below are captured from Cisco testimonial which presents a sample of a business request: This example from Cisco testimonial shows that RedSeal may simulate an open port from a particular network.

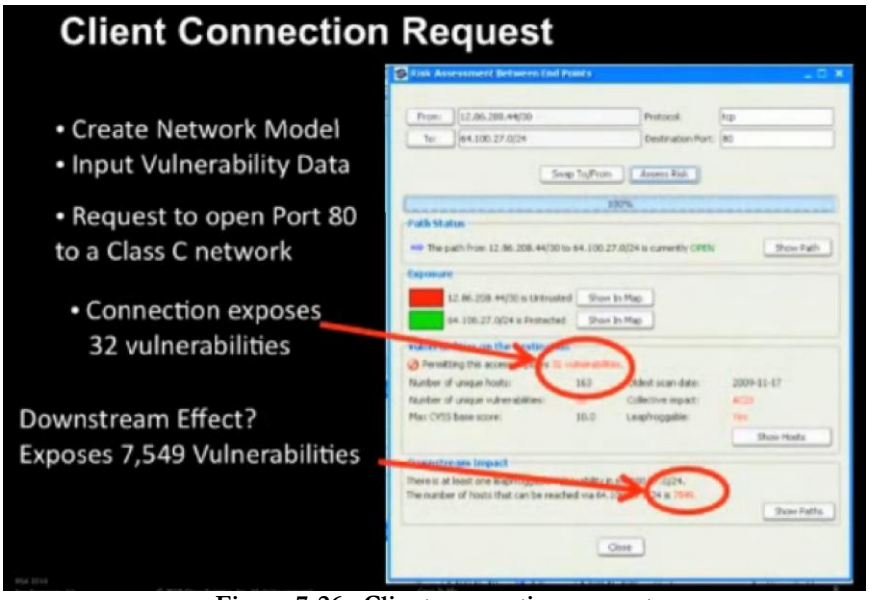


Figure 7-26 : Client connection request

And the result from this request under a defined network infrastructure topology:

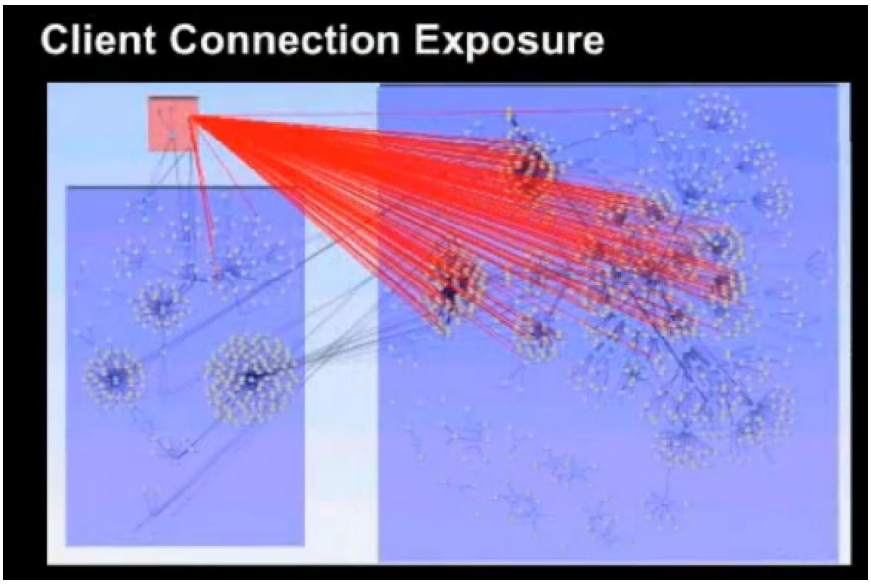


Figure 7-27 : Client connection exposure

As a conclusion, RedSeal provides a tool that fits to the task of evaluating and measuring the impact of changes in network infrastructure. Unfortunately, there is little information about the way the tool works. There is only business brochures (or videos) explaining that RedSeal platform perform network risk security management and help organizations. So, for example, we do not know exactly how they collect information from network devices, what kind of changes it is possible to do (open/close port, change destination, change protocol, add/delete device, etc.), how it performs leapfrog's attack and so on.

8. Conclusion

This report presents D2.1: (T0 + 3) Intrusion detection, prevention and reaction simulation systems (IDS). The state of the art for intrusion detection systems is given in terms of both R&D reports and commercially available products. The main techniques are focused on anomaly detection and signature detection types.

Various attack types in particular DDoS attacks are investigated in detail along with complex attacks. Following the investigation of attack types, detection and countermeasure techniques are covered. Specially Bayesian methods are also examined, and are considered in the detection block. The requirements for an attack simulation environment are studied. DETER testbed will be an important test bed for the attack detection and countermeasure systems throughout the project.

Until coordinating for real data, the test-bed will be a main component of research and development.
In terms of network simulation systems, OPNET is a good candidate for future tasks about security activity simulation as it has feature specialized in cyber security (attack and countermeasure simulation). EXata/Cyber from Scalable Network Technologies seems promising as it has been designed for testing network resiliency to cyber-attacks. It is needed to check if testing countermeasure is possible with this software simulation. Other tools should be hard to reuse because they are too much oriented on low network layers or solely dedicated to wireless networks (e.g. GloMoSim), or not usable for security purpose. Others, such as NS-3, appear too much complex in terms of modeling to be considered as potential candidates in an operational perspective.

Regarding the simulation of security impact, RedSeal and Skybox could provide interesting but not sufficient features. E.g. they are focused on vulnerabilities and not in attack detection. In the other hand, they would probably require the use of unnecessary functions for the purpose of ADAX. Concretely, one can imagine the following scenario: a device developed by ADAX partners detects an ongoing complex attack and sends an alert notification to the countermeasure engine. Once this one recommends remediation actions, then it sends a request to RedSeal or Skybox to assess the action impacts if enforced. Difficulties come at this stage for 1) it is not clear if an interface can be made for such requests, 2) these tools have their own workflow and probably require information not available by an automated process (i.e. the ADAX countermeasure engine). For RedSeal, there is a strict and heavy workflow to comply with whereas only the 1st step gets interesting for the project purpose. Another option, maybe not acceptable, would be to think about external verifications made independently from any ADAX device. Once simulation results are available, the operator analyzes them, writes comments on the countermeasure device and selects actions to enforce.

9. References

[1] Peng Ning and Sushil Jajodia.Intrusion Detection Techniques. John Wiley and Sons, Inc., 2004.
[2] ve Haque S. S. Faysel, M. A. Towards cyber defense: Research in intrusion detection and intrusion prevention systems. In International Journal of Computer Science and Network Security, volume 10, 2010.
[3] H Debar. An introduction to intrusion-detection systems. In Proceedings of Connect'2000, 2000.
[4] M. Roesch. Snort - lightweight intrusion detection for networks. In Proceedings of the 13th USENIX conference on System administration, 1999.
[5] Smaha S.E. Haystack: An intrusion detection system. In Proceedings of the IEEE Fourth Aero-space Computer Security Applications Conference, 1988.
[6] Inc. Network Flight Recorder. Network flight recorder, December 2011.
[7] Jacobson V McCanne S., Leres C. libpcap, 1994.
[8] Valdes A Anderson D., Frivold T. Next-generation intrusion detection expert system (nides). In Technical Report SRI-CSL-95-07, SRI International, Computer Science Lab, 1995.
[9] Jagannathan R. Lunt, T. F. A prototype real-time intrusion-detection expert system. In IEEE Symposium on Security and Privacy, p. 59, 1988.
[10] Javitz H. Tamaru A. Valdes A. Anderson D., Lunt T. Detecting unusual program behavior using the statistical component of the next-generation intrusion detection expert system (nides). In SRI International Computer Science Laboratory Technical Report SRI-CSL-95-06, 1995.
[11] Sebring et al. Expert systems in intrusion detection: A case study. In Proceedings of the11th National Computer Security Conference, 1988.
[12] Neumann PG. Porras PA. Emerald: Event monitoring enabling responses to anomalous live disturbances. In Proceedings of the 20th National Information Systems Security Conference, pp. 353-365, 1997.
[13] Ramstedt P. Dowell C. The computer watch data reduction tool. In Proc. 13th National Computer Security Conf., Washington, DC, pp. 99-108, 1990.
[14] Kemmerer R. Eckmann S., Vigna G. Statl: An attack language for state-based intrusion detection. In Proceedings of the ACM Workshop on Intrusion Detection Systems, 2000.
[15] Porras P. Stat - a state transition analysis tool for intrusion detection. In Master's thesis, Computer Science Department, University of California, 1992.
[16] Ilgun K. Ustat: A real-time intrusion detection system for UNIX. In Master's thesis, Computer Science Department, University of California, 1992.
[17] Kemmerer RA Vigna G. Netstat: A network-based intrusion detection approach. In Proceedings of the 14th Annual Computer Security Applications Conference, pp.25, 1998.
[18] Wenke Lee, S.J. Stolfo, and K.W. Mok. A data mining framework for building intrusion detection models. In Security and Privacy, 1999. Proceedings of the 1999 IEEE Symposium on, pages 120 {132, 1999.
[19] Bin Y. Ge S. Qiao Y., Xin XW. Anomaly intrusion detection method based on hmm. In IEEE Electronic Letters Online No: 20020467, 2002.
[20] Schwartzbaxd A. Ghosh AK. A study in using neural networks for anomaly and misuse detection. In Proceedings of the 8th USENIX Security Symposium, 1999.
[21] Reed JH. Simonian RP. Fox KL., Henning RR. A neural network approach towards intrusion detection. In Proceedings of the 13th National Computer Security Conference, Washington, D.C. Gaithersburg, MD: NIST, 125-134, 1990.
[22] Kianie M. Mohajerani M., Moeini A. Nds: A neuro-fuzzy intrusion detection system. In Proceedings of the 10th IEEE International Conference on Electronics, Circuits and Systems, pp348-351, 2000.
[23] Manson G.A Abouzakhar N.S. Networks security measures using neuro-fuzzy agents. In Journal of Information Management and Computer Security. 11 (1), pp.33-38, 2003.
[24] Saxton LV. Yao JT, Zhao SL. A study on fuzzy intrusion detection. In Proceedings of SPIE Vol. 5812, Data Mining, Intrusion Detection, Information Assurance, And Data Networks Security, Orlando, Florida, USA: SPIE, Bellingham, WA, pp. 23-30, 2005.
[25] Marrakchi Zakia Puttini, Ricardo S. and Ludovic M. A Bayesian classification model for real-time intrusion detection. In API Conference, pp. 150-162, 2003.
[26] Ng RT. Sander J. Breunig MM., Kriegel HP. Lof: identifying density-based local outliers. In Proceedings of the 2000 ACM SIGMOD international conference on Management of data, Dallas, Texas, United States, pp.93-104, 2000.
[27] Boukelif A. Faraoun KM. Neural networks learning improvement using the k-means clustering algorithm to detect network intrusions. International Journal of Computational Intelligence 3;2; p.28-36, 2007.

[28] Ozgur J. Srivastava A. Lazarevic, L. Ertoz and V. Kumar. A comparative study of anomaly detection schemes in network intrusion detection. In Proceedings of SIAM Conf. Data Mining, 2003.
[29] James P. Anderson. Computer Security Threat Monitoring and Surveillance. 1980.
[30] Gartner “Magic Quadrant for Network Intrusion Prevention Systems” by Greg Young and John Pescatore. December 6, 2010 (ID Number: G00208628)
[31] Anonymous. Top 125 network security tools. 2011.
[32] Anonymous. Check point IPS engine architecture, 2011.
[33] Anonymous. Secure networks for process control, 2011.
[34] P. Bicknell and J. Hung. Validation report HP TippingPoint intrusion prevention systems, 2011.
[35] Anonymous. Installation and configuration guide for IPS deployments of IBM Proventia net-work IPS on crossbeam x-series systems, 2008.
[36] Anonymous. Snort user’s manual, 2009.
[37] Anonymous. Http://www.ossec.net. 2011.
[38] O. Leon J. Hernandez-Serrano and M. Soriano. Modeling the lion attack in cognitive radio net-works. EURASIP Journal on Wireless Communications and Networking, 2011:1{10, 2011.
[39] F. G. Marmol and G. M. Perez. Security threats scenarios in trust and reputation models for distributed systems. Computers and Security, 28:545{556, 2009.
[40] G. Thamilarasu and R. Sridhar, editors. Exploring Cross-Layer Techniques for Security: Challenges and Opportunities in Wireless Networks, 2007.
[41] J. H. Serrano O. Leon and M. Soriano. A new cross-layer attack to TCP in cognitive radio net-works, 2009.
[42] Denial-of-service developments. Technical report, CERT Coordination Center, 2000.
[43] Usman Tariq, Manpyo Hong, and Kyung-suk Lhee. A Comprehensive Categorization of DDoS Attack and DDoS Defense Techniques *.Program, (Mic):1025 { 1036, 2006.
[44] Jelena Mirkovic and Peter Reiher. A taxonomy of DDoS attack and DDoS defense mechanisms. SIGCOMM Comput. Commun. Rev., 34:39{53, April 2004.
[45] SM. Specht. Distributed denial of service: Taxonomies of attacks, tools, and countermeasures. In Citeseer. Proceedings of the 17th International Conference, 2004.
[46] C Douligeris and A Mitrokotsa. DDoS attacks and defense mechanisms: a classification. Proceedings of the 3rd IEEE International Symposium on Signal Processing and Information Technology IEEE Cat No03EX795, pages 190{193, 2004.
[47] David Champagne and Rb Lee. Scope of DDoS countermeasures: taxonomy of proposed solutions and design goals for real-world deployment. On Systems and Information Security SSI, 2006.
[48] Ratul Mahajan, Steven M. Bellovin, Sally Floyd, John Ioannidis, Vern Paxson, and Scott Shenker. Controlling high bandwidth aggregates in the network. ACM Computer Communication Review, 32:62{73, 2002.
[49] Angelos D Keromytis, Vishal Misra, and Dan Rubenstein. Sos: Secure overlay services. Electrical Engineering, 32(4):61{72, 2002.
[50] Abbass Asosheh and Naghmeh Ramezani. A Comprehensive Taxonomy of DDoS Attacks and Defense Mechanism Applying in a Smart Classification. WSEAS Transactions on Computers, 7(4):281{290, 2008.
[51] Oliver Spatscheck and Larry L Peterson. Defending Against Denial of Service Attacks in Scout, pages 59{72. Usenix and Association for Computing Machinery, 1999.
[52] R Lee, D Karig, J McGregor, and Zhijie Shi. Enlisting hardware architecture to thwart malicious code injection. Security in Pervasive Computing, pages 237{252, 2004.
[53] Kevin J. Houle. Trends in denial of service attack. Technical report, CERT Coordination Center, Carnegie Mellon Software Engineering Institute., 2001.
[54] Sven Dietrich, Neil Long, and David Dittrich. Analyzing Distributed Denial of Service Tools: The Shaft Case, pages 329{339. 2000.
[55] David Karig Lee and Ruby. Remote denial of service attacks and countermeasures. Technical report, Princeton University Department of Electrical Engineering, 2001.
[56] Wei Wang and Sylvain Gombault. Efficient detection of DDoS attacks with important at-tributes. In Mohamed Jmaiel and Mohamed Mosbah, editors, CRiSIS, pages 61{67. IEEE, 2008.
[57] Darpa intrusion detection evaluation, 1999.
[58] Bin Xiao, Wei Chen, Yanxiang He, and E.H.-M. Sha. An active detecting method against syn-floding attack. In Parallel and Distributed Systems, 2005. Proceedings11th International Conference on, volume 1, pages 709 {715 Vol. 1, July 2005.
[59] Xie Chuiyi, Zhang Yizhi, Bai Yuan, Luo Shuoshan, and Xu Qin. A distributed intrusion detection system against flooding denial of services attacks. In Advanced Communication Technology (ICACT), 2011 13th International Conference on, pages 878 {881, Feb. 2011.

[60] J. Brustoloni. Protecting electronic commerce from distributed denial-of-service attacks. Pages 553-561. 11th international conference on World Wide Web, May 2002.
[61] H Burch and B Cheswick. Tracing Anonymous Packets to Their Approximate Source, pages 313-322. 2000.
[62] Aleya Hussain, Stephen Schwab, Roshan Thomas, and Sonia Fahmy. DDoS experiment methodology. October, pages 8{14, 2006.
[63] Abdulkawi A., Saleh T. S., Khattab S., and Farag I. (2012, May). Anti-jamming defense in wireless networks using channel hopping and error correcting code. In 8th International Conference on Informatics and Systems (INFOS), (pp. 12-17).
[64] Barros J. and Rodrigues M. R. D. (2006, July). Secrecy capacity of wireless channels. In Proc. IEEE International Symposium on Information Theory, Seattle, WA, (pp. 356–360).
[65] Bloch M.; et al. (2008). Wireless Information-Theoretic Security. IEEE Transactions on Information Theory, (pp. 2515–34).
[66] Chang Q., Zhang Y. P. and Qin L. L. (2010, June). A node authentication protocol based on ECC in WSN. In International Conference on Computer Design and Applications (ICCDA), (pp. 606-609).
[67] Chen Z., Nan X. H. (2006). CPK identity authentication. In Beijing, China: National Defense Indus-try Press.
[68] Csiszár I., Korner J. (1978, May). Broadcast Channels with Confidential Messages. IEEE Transactions on Information Theory, 24, (pp. 339–348).
[69] Diffie, W., Hellman, M. (1976). New directions in cryptography. IEEE Transactions on Information Theory, 22(6), (pp. 644–654).
[70] Dong L., Han Z., Petropulu A. P., and Poor H. V. (2008, Sept.). Secure wireless communications via cooperation. In Proceedings of the 46th Annual Allerton Conference on Communications, Control, Computing, Monticello, IL, (pp. 1132–1138).
[71] Dong L., Han Z., Petropulu A. P., and Poor H. V. (2009, April). Amplify-and-forward based cooperation for secure wireless communications. In Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, Taipei, Taiwan, (pp. 2613–2616).
[72] Ekrem E. and Ulukus S. (2009). Cooperative secrecy in wireless communications. In Securing Wireless Communications at the Physical Layer, R. Liu and W. Trappe, Eds. New York: Springer-Verlag.
[73] Franklin M., Boneh D. (2001). Identity based encryption from weil pairing. Proceedings of CRYP-TO 2001, Berlin: Springer Verlag, (pp. 213-239).
[74] Goel S. and Negi R. (2008, June). Guaranteeing secrecy using artificial noise. IEEE Transactions on Wireless Communications, 7(6), (pp. 2180–2189).
[75] Huang Y. and Palomar D.P. (2010). Rank-constrained separable semi definite programming with applications to optimal beamforming. IEEE Transactions on Signal Processing, 58(2), (pp. 664–678).
[76] Jorgensen M. L., Yanakiev B. R., Kirkelund F. E., Popovski P., Yomo H., and Larsen T. (2007, Nov.). Shout to secure: Physical-layer wireless security with known interference. In Proceedings of IEEE GLOBECOM, Washington, DC, (pp. 33–38).
[77] Karas D. S., Karagiannidis G. K., and Schober R. (2011, Sept.). Neural network based PHY-layer key exchange for wireless communications. In IEEE 22nd International Symposium on Personal In-door and Mobile Radio Communications (PIMRC), (pp. 1233-1238).
[78] Khisti A. and Wornell G.W. (2007). Secure transmission with multiple antennas: The MIMOME channel," IEEE Transactions of Information Theory, available online, http://arxiv.org/abs/0708.4219.
[79] Kurita S., Komoriya K., and Uda R. (2012, Mar.) Privacy protection on transfer system of auto-mated teller machine from brute force attack. In 26th International Conference on Advanced Information Networking and Applications Workshops (WAINA), (pp. 72-77).
[80] Lai L., Liang Y. and Du W. (2012). Cooperative key generation in wireless networks. IEEE Journal on Selected Areas in Communications, 30(8), (pp. 1578-1588).
[81] Li W., Ghogho M., Chen B. and Xiong C. (2012, Oct.). Secure communication via sending artificial noise by the receiver: Outage secrecy capacity/region analysis. IEEE Communication Letters, 16(10), (pp. 1628 –1631).
[82] Li, Z., Trappe W. and Yates, R. (2007, March). Secret communication via multiantenna transmis-sion. In Proceedings of 41st CISS, Baltimore, MD, (pp. 905–910).
[83] Liang Y., Poor H. V., and Shamai S., (2008, June). Secure communication over fading channels. IEEE Transactions on Information Theory, 54(6), (pp. 2470–2492).
[84] Liao W. C., Chang T. H., Ma W. K., and Chi C. Y. (2010, March). Joint transmit beamforming and artificial noise design for QoS discrimination in wireless downlink. In 2010 IEEE International Conference on Acoustics Speech and Signal Processing, (pp. 2562 –2565).

[85] Mingyan L., Koutsopoulos I., and Poovendran R. (2010, Aug.). Optimal jamming attack strategies and network defense policies in wireless sensor networks. IEEE Transactions on Mobile Computing, 9(8), (pp. 1119-1133).
[86] Nan X.H and Chen Z. (2006). Identifier-based private key generating method and device, WO Pa-tent WO/2006/074,611.
[87] Negi R. and Goel S. (2005, Sept.). Secret communications using artificial noise. In Proceedings of IEEE Vehicular Technology Conference, Dallas, TX, (pp. 1906–1910).
[88] Oggier F. and Hassibi B. (2008, July). The secrecy capacity of the MIMO wiretap channel. In Proceedings of IEEE International Symposium on Information Theory, Toronto, ON, Canada, (pp. 524–528).
[89] Ozharar S., Reilly D. R., Wang S. X., Kanter G. X. and Kumar P. (2011, July). Two dimensional optical code-division modulation with quantum-noise aided encryption for applications in key distribution. Journal of Lightwave Technology, 29(14), (pp.2081-2088).
[90] Prabhu V. and Rodrigues M. (2011). On wireless channels with m-antenna eavesdroppers: Characterization of the outage probability and outage secrecy capacity. IEEE Transactions on Information Forensics Security, 99.
[91] Ren K., Su H., and Wang Q. (2011, Aug.). Secret key generation exploiting channel characteristics in wireless communications. In IEEE Wireless Communications} 18(4), (pp. 6-12).
[92] Shamir A. (1984). Identity-based cryptosystems and signature schemes. In Proc of CRYPTO'84. Berlin: Springer Verlag. (pp. 47-53), Springer.
[93] Shannon C. E. (1949, Oct.). Communication Theory of Secrecy Systems. The Bell System Tech-nical Journal, 28(4), (pp. 656-715).
[94] Soosahabi R. and Naraghi-Pour M. (2012, Aug.), Scalable PHY-layer security for distributed detection in wireless sensor networks. In IEEE Transactions on Information Forensics and Security, 7(4), (pp. 1118-1126).
[95] Swindlehurst A. L. (2009, Apr.). Fixed SINR solutions for the MIMO wiretap channel. In Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, Taipei, Taiwan, (pp. 2437–2440).
[96] Wyner, A. D. (1975). The Wire-tap Channel. The Bell System Technical Journal, 54, (pp. 1355–1387).
[97] Yang Y., Wang W., Zhao H., and Zhao L. (2012). Transmitter beamforming and artificial noise with delayed feedback: Secrecy rate and power allocation. In Journal of Communications and Networks, 14(4), (pp. 374-384).
[98] Zhuo C. , Fan H., Liang H. (2005, Apr.). A new authentication and key exchange protocol in WLAN. In International Conference on Information Technology: Coding and Computing, ITCC, 2, (pp. 552-556).
[99] J. Mitola III and G.Q.,Jr. Maguire, Cognitive radio: making software radios more personal Communications, IEEE , vol.6, no.4, pp.13-18, Aug 1999.IEEE NETWORK MAGAZINE, VOL. X, NO. X, MAY 2013 9
[100] T.C. Clancy and N. Goergen, Security in Cognitive Radio Networks: Threats and Mitigation Cognitive Radio Oriented Wireless Networks and Communications, 2008. Crown-Com 2008. 3rd International Conference on, vol., no., pp.1-8, 15-17 May 2008.
[101] I.F. Akyildiz, W-Y. Lee, M.C. Vuran, and S. Mohanty, NeXt Generation/Dynamic Spectrum Access/Cognitive Radio Wireless Networks A Survey, Computer Networks, vol.50, pp.2127-2159, May 2006.
[102] B.A. Fette, Cognitive Radio Technology, 1st ed. vol.1. Massachusetts: Elsevier, 622, pp. 223-224.
[103] G. Baldini and et al., Security Aspects in Software Defined Radio and Cognitive Radio Networks: A Survey and A Way Ahead, Communications Surveys & Tutorials, IEEE , vol.14, no.2, pp.355-379, Second Quarter 2012.
[104] L. Husheng and H. Zhu, Dogfight in Spectrum: Combating Primary User Emulation Attacks in Cognitive Radio Systems Part II: Unknown Channel Statistics, Wireless Communications, IEEE Transactions on , vol.10, no.1, pp.274-283, January 2011.
[105] W. Wang et al., Cross-Layer Attack and Defense in Cognitive Radio Networks Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE, pp.1-6, 6-10 Dec. 2010.
[106] I . Aad, J.-P. Hubaux, and E .W. Knightly, Denial of service resilience in ad hoc networks Proceedings of the 10th Annual International Conference on Mobile Computing and Networking (MobiCom) Philadelphia, PA, September, 2004.
[107] C. Cordeiro, K. Challapali, D. Birru, and N. S. Shankar, IEEE 802.22: The first worldwide wireless standard based on cognitive radios in Proc. of First IEEE International Symposium on Dynamic Spectrum Access Networks (DyS-PAN05), pp. 328-337, 8-11 November 2005.
[108] D. Niyato and E. Hossain, Competitive pricing for spectrum sharing in cognitive radio networks: Dynamic game, inefficiency of Nash equilibrium, and collusion, IEEE Journal on Selected Areas in Communications, vol. 26, no. 1, pp. 192-202, January 2008.
[109] Y. Guosen and W. Xiaodong, Antijamming Coding Techniques with Application to Cognitive Radio IEEE Transactions on Wireless Comm., vol.8, no.12, pp.5996-6007, 2009.

[110] Li W., Ghogho M., Chen B. and Xiong C. (2012, Oct.). Secure communication via sending artificial noise by the receiver: Outage secrecy capacity/region analysis. IEEE Communication Letters, 16(10), (pp. 1628 –1631).
[111] Wyner, A. D. (1975). The Wire-tap Channel. The Bell System Technical Journal, 54, (pp. 1355–1387).
[112] Shannon C. E. (1949, Oct.). Communication Theory of Secrecy Systems. The Bell System Technical Journal, 28(4), (pp. 656-715).
[113] Csiszár I., Korner J. (1978, May). Broadcast Channels with Confidential Messages. IEEE Trans-actions on Information Theory, 24, (pp. 339–348).
[114] Diffie, W., Hellman, M. (1976). New directions in cryptography. IEEE Transactions on Information Theory, 22(6), (pp. 644–654).
[115] Noubir, G. (2004). "On Connectivity in Ad Hoc Network Under Jamming Using Directional Antennas and Mobility," 2nd Int. Conf. Wired and Wireless Internet Commun, (pp. 54–62.).
[116] Li, Z., Trappe W. and Yates, R. (2007, March). Secret communication via multiantenna transmission. In Proceedings of 41st CISS, Baltimore, MD, (pp. 905–910).
[117] Khisti A. and Wornell G.W. (2007). Secure transmission with multiple antennas: The MIMOME channel," IEEE Transactions of Information Theory, 56(7), (pp. 3088-3104).
[118] Oggier F. and Hassibi B. (2008, July). The secrecy capacity of the MIMO wiretap channel. In Proceedings of IEEE International Symposium on Information Theory, Toronto, ON, Canada, (pp. 524–528).
[119] Bloch M.; et al. (2008). Wireless Information-Theoretic Security. IEEE Transactions on Information Theory, (pp. 2515–34).
[120] Liang Y., Poor H. V., and Shamai S., (2008, June). Secure communication over fading channels. IEEE Transactions on Information Theory, 54(6), (pp. 2470–2492).
[121] Jorgensen M. L., Yanakiev B. R., Kirkelund F. E., Popovski P., Yomo H., and Larsen T. (2007, Nov.). Shout to secure: Physical-layer wireless security with known interference. In Proceedings of IEEE GLOBECOM, Washington, DC, (pp. 33–38).
[122] Dong L., Han Z., Petropulu A. P., and Poor H. V. (2008, Sept.). Secure wireless communications via cooperation. In Proceedings of the 46th Annual Allerton Conference on Communications, Control, Computing, Monticello, IL, (pp. 1132–1138).
[123] Dong L., Han Z., Petropulu A. P., and Poor H. V. (2009, April). Amplify-and-forward based cooperation for secure wireless communications. In Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, Taipei, Taiwan, (pp. 2613–2616).
[124] Ekrem E. and Ulukus S. (2009). Cooperative secrecy in wireless communications. In Securing Wireless Communications at the Physical Layer, R. Liu and W. Trappe, Eds. New York: Springer-Verlag.
[125] Negi R. and Goel S. (2005, Sept.). Secret communications using artificial noise. In Proceedings of IEEE Vehicular Technology Conference, Dallas, TX, (pp. 1906–1910).
[126] Goel S. and Negi R. (2008, June). Guaranteeing secrecy using artificial noise. IEEE Transactions on Wireless Communications, 7(6), (pp. 2180–2189).
[127] Swindlehurst A. L. (2009, Apr.). Fixed SINR solutions for the MIMO wiretap channel. In Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, Taipei, Taiwan, (pp. 2437–2440).
[128] Prabhu V. and Rodrigues M. (2011). On wireless channels with m-antenna eavesdroppers: Characterization of the outage probability and outage secrecy capacity. IEEE Transactions on Information Forensics Security, 99.
[129] Liao W. C., Chang T. H., Ma W. K., and Chi C. Y. (2010, March). Joint transmit beamforming and artificial noise design for QoS discrimination in wireless downlink. In 2010 IEEE International Conference on Acoustics Speech and Signal Processing, (pp. 2562 –2565).
[130] Lai L., Liang Y. and Du W. (2012). Cooperative key generation in wireless networks. IEEE Journal on Selected Areas in Communications, 30(8), (pp. 1578-1588).
[131] Shamir A. (1984). Identity-based cryptosystems and signature schemes. In Proc of CRYPTO'84. Berlin: Springer Verlag. (pp. 47-53), Springer.
[132] Franklin M., Boneh D. (2001). Identity based encryption from weil pairing. Proceedings of CRYPTO 2001, Berlin: Springer Verlag, (pp. 213-239).
[133] Chen Z., Nan X. H. (2006). CPK identity authentication. In Beijing, China: National Defense Industry Press.
[134] Nan X.H and Chen Z. (2006). Identifier-based private key generating method and device, WO Pa-tent WO/2006/074,611.
[135] Kurita S., Komoriya K., and Uda R. (2012, Mar.) Privacy protection on transfer system of auto-mated teller machine from brute force attack. In 26th International Conference on Advanced Information Networking and Applications Workshops (WAINA), (pp. 72-77).
[136] Chang Q., Zhang Y. P. and Qin L. L. (2010, June). A node authentication protocol based on ECC in WSN. In International Conference on Computer Design and Applications (ICCDA), (pp. 606-609).

[137] Ozharar S., Reilly D. R., Wang S. X., Kanter G. X. and Kumar P. (2011, July). Two dimensional optical code-division modulation with quantum-noise aided encryption for applications in key distribution. Journal of Lightwave Technology, 29(14), (pp.2081-2088).
[138] Abdulkawi A., Saleh T. S., Khattab S., and Farag I. (2012, May). Anti-jamming defense in wire-less networks using channel hopping and error correcting code. In 8th International Conference on Informatics and Systems (INFOS), (pp. 12-17).
[139] Yang Y., Wang W., Zhao H., and Zhao L. (2012). Transmitter beamforming and artificial noise with delayed feedback: Secrecy rate and power allocation. In Journal of Communications and Net-works, 14(4), (pp. 374-384).
[140] Mingyan L., Koutsopoulos I., and Poovendran R. (2010, Aug.). Optimal jamming attack strategies and network defense policies in wireless sensor networks. IEEE Transactions on Mobile Computing, 9(8), (pp. 1119-1133).
[141] Huang Y. and Palomar D.P. (2010). Rank-constrained separable semi definite programming with applications to optimal beamforming. IEEE Transactions on Signal Processing, 58(2), (pp. 664–678).
[142] Zhuo C., Fan H., Liang H. (2005, Apr.). A new authentication and key exchange protocol in WLAN. In International Conference on Information Technology: Coding and Computing, ITCC, 2, (pp. 552-556).
[143] Ren K., Su H., and Wang Q. (2011, Aug.). Secret key generation exploiting channel characteristics in wireless communications. In IEEE Wireless Communications} 18(4), (pp. 6-12).
[144] Karas D. S., Karagiannidis G. K., and Schober R. (2011, Sept.). Neural network based PHY-layer key exchange for wireless communications. In IEEE 22nd International Symposium on Personal In-door and Mobile Radio Communications (PIMRC), (pp. 1233-1238).
[145] Soosahabi R. and Naraghi-Pour M. (2012, Aug.), Scalable PHY-layer security for distributed detection in wireless sensor networks. In IEEE Transactions on Information Forensics and Security, 7(4), (pp. 1118-1126).
[146] P. Bedi, V. Gandotra, A. Singhal, H. Narang, and S. Sharma. Optimal Countermeasures Identification Method: A New Approach in Secure Software Engineering. *European Journal of Scientific Research*, 55(4):527-537, 2011.
[147] S. Bistarelli, F. Fioravanti, and P. Peretti. Using CP-nets as a Guide for Countermeasure Selection. In *ACM Symposium on Applied Computing*, pages 300-308, 2007.
[148] S. Bistarelli, F. Fioravanti, P. Peretti, and I. Trubitsyna. Modeling and selecting countermeasures using CP-nets and Answer Set Programming. In *23rd Italian Convention of Computational Logic*, 2008.
[149] C. A. Carver and U. W. Pooch. An intrusion response taxonomy and its role in automatic intrusion response. In *IEEE Systems, Man, and Cybernetics Information Assurance and Security Workshop*, June 2000.
[150] H. Cavusoglu, B. Mishra, and S. Raghunathan. A Model for Evaluating It Security Investment. *Communications of the AMC*, 47(7):87-92, 2004.
[151] H. Debar, Y. Thomas, F. Cuppens, and N. Cuppens-Boulahia. Enabling automated threat response through the use of a dynamic security policy. *Journal in Computer Virology (JCV)*, 3:195-210, August 2007.
[152] C. Duan and J. Cleland-Huang. Automated Safeguard Selection Strategies. In *CTI Research Symposium*, 2006.
[153] Y. Ferenc and Y. Salim. A Game Theory Based Risk and Impact Analysis Method for Intrusion Defense Systems. In *International Conference on Computer Systems and Applications (AICCSA)*, pages 975-982, 2009.
[154] E. A. Fisch. *A Taxonomy and implementation of automated responses to intrusive behavior*. PhD thesis, Texas A&M University, 1996.
[155] C. Irvine and T. Levin. Toward a Taxonomy and Costing Method for Security Services. In *Computer Security Applications Conference*, pages 163-168, 1999.
[156] A. A. E. Kalam, R. E. Baida, P. Balbiani, S. Benferhat, F. Cuppens, Y. Deswarte, A. Miège, C. Saurel, and G. Trouessin. Organization based access control. In *the fourth IEEE International Workshop on Policies for Distributed Systems and Networks (Policy)*, Lake Come, Italy, June 2003.
[157] K. S. Killourhy, R. A. Maxion, and K. M. C. Tan. A defense-centric taxonomy based on attack manifestations. In *Proceedings of the 2004 International Conference on Dependable Systems and Networks (DSN'04)*, 2004.
[158] A. Kim, J. Luo, and M. Kang. Security Ontology for Annotating Resources. In *Research Lab, NRL Memorandum Report*, pages 1483-1499, 2005.
[159] J. Mirkovic and P. Reiher. A Taxonomy of DDoS Attack and DDoS Defense Mechanisms. In *ACM SIGCOMM Computer Communication Review*, volume 34, pages 39-53, 2004.
[160] T. Neubauer and M. Pehn. Workshop-based Security Safeguard Selection with AURUM. *International Journal on Advances in Security*, 3(3-4):123-134, 2010.
[161] T. Neubauer, C. Stummer, and E. Weippl. Workshop-based Multiobjective Security Safeguard Selection. In *First International Conference on Availability, Reliability and Security (ARES)*, pages 1-8, 2006.

[162] T. Norman. *Risk Analysis and Security Countermeasure Selection*. CRC Press Taylor & Francis Group, 2010.
[163] M. Papadaki, S. Furnell, B. Lines, and R. Reynolds. A response oriented taxonomy of IT system intrusions. In *Proceedings of Euromedia*, Italy, 2002.
[164] M. Papadaki and S. M. Furnell. Informing the decision process in an automated intrusion response system. *Information Security Technical Report*, 10:150-161, 2005.
[165] M. Schumacher. *Security engineering with patterns: origins, theoretical model, and new applications*. Springer-Verlag New York Inc, 2003.
[166] P. Smith, A. Schaeffer-Filho, A. Azman, M. Scoller, N. Kheir, A. Mauthe, and D. Hutchison. Strategies for network resilience: Capitalizing on policies. In *Proceedings of the fourth International Conference on Autonomous Infrastructure, Management and Security (AIMS)*, 2010.
[167] N. Stakhanova, S. Basu, and J. Wong. Taxonomy of intrusion response systems. *International Journal of Information and Computer Security*, 1(1/2):169-184, 2007.
[168] A. Talib, R. Atan, R. Abdullah, and M. Azmi. Security Ontology Driven Multi Agent System Architecture for Cloud Data Storage Security: Ontology Development. *International Journal of Computer Science and Network Security*, 12(5):63-72, 2012.
[169] Y. Thomas. *Policy-Based Response to Intrusions Through Context Activation*. PhD thesis, Ecole Nationale Supérieure des Télécommunications de Bretagne, 2007.
[170] H. Venter and J. Eloff. A Taxonomy for Information Security Technologies. *Computers and Security*, 22:299-307, 2003.
[171] H. Wang and C. Wang. Taxonomy of Security Considerations and Software Quality. *Communications of the ACM*, 46(6):75-78, 2003.
[172] Classification and regression trees. Monterey, California, 1984.
[173] Paul Barford, Jeery Kline, David Plonka, and Amos Ron. A signal analysis of network traffic anomalies. Proceedings of the second ACM SIGCOMM Workshop on Internet measurement IMW 02, page 71, 2002.
[174] Irad Ben-Gal. Bayesian networks. Networks, 2007.
[175] Lee D. C. Fast traffic anomalies detection using snmp mib correlation analysis. In Proceedings of International Conference on Advanced Communication Technology (ICACT), 2009.
[176] Jin Cao, W.S. Cleveland, Yuan Gao, K. Je ay, F.D. Smith, and M. Weigle. Stochastic models for generating synthetic http source traffic. In INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies, volume 3, pages 1546 {1557 vol.3, March 2004.
[177] Alberto Carrascal, Jorge Couchet, Enrique Ferreira, and Daniel Manrique. Anomaly detection using prior knowledge: application to TCP/IP traffic. Learning, 217:139{148, 2006.
[178] Guillaume Dewaele, Kensuke Fukuda, Pierre Borgnat, Patrice Abry, and Kenjiro Cho.Ex-tracting hidden anomalies using sketch and non-Gaussian multi resolution statistical detection procedures, pages 145-152. ACM Press, 2007.
[179] S Floyd and V Paxson. Difficulties in simulating the internet. IEEE/ACM Transactions on Net-working, 9(4):392-403, 2001.
[180] Sally Floyd and Eddie Kohler. Internet research needs better models. ACM SIGCOMM Computer Communication Review, 33(1):29{34, 2003.
[181] Romain Fontugne, Toshio Hirotsu, and Kensuke Fukuda. An image processing approach to traffic anomaly detection. Proceedings of the 4th Asian Conference on Internet Engineering AINTEC 08, page 17, 2008.
[182] D A N Gorton. Extending intrusion detection with alert correlation and intrusion tolerance. October, 2003.
[183] J W Haines, L M Rossey, R P Lippmann, and R K Cunningham. Extending the Darpa online intrusion detection evaluations. Proceedings DARPA Information Survivability Conference and Exposition II DISCEX01, 1:35{45, 1999.
[184] Wes Hardaker. Justification and requirements for a national DDoS defense technology evaluation facility. Network, 2002.
[185] Sourcefire homepages. Sourcefire IPS. http://www.sourcefire.com/content/next-generation-intrusion-prevention-system-ngips, 2010.
[186] Alefiya Hussain, Stephen Schwab, Roshan Thomas, and Sonia Fahmy. DDoS experiment methodology. October, pages 8{14, 2006.
[187] Lawrence Berkeley National Laboratory. Bro intrusion detection system. http://bro-ids.org/, 2011.
[188] Anukool Lakhina, Mark Crovella, and Christophe Diot. Mining anomalies using traffic feature distributions. ACM SIGCOMM Computer Communication Review, 35(4):217, 2005.
[189] Keunsoo Lee, Juhyun Kim, Ki Hoon Kwon, Younggoo Han, and Sehun Kim. DDoS attack detection method using cluster analysis. Expert Systems with Applications, 34(3):1659{1665, 2008.

[190] Youngseok Lee and B. Mukherjee. Traffic engineering in next-generation optical networks. Communications Surveys Tutorials, IEEE, 6(3):16 {33, quarter 2004.
[191] R P Lippmann, D J Fried, I Graf, J W Haines, K R Kendall, D McClung, D Weber, S E Webster, D Wyschogrod, R K Cunningham, and et al. Evaluating intrusion detection systems: the 1998 Darpa online intrusion detection evaluation. Proceedings DARPA Information Survivability Conference and Exposition DISCEX00, 2(c):12{26, 2000.
[192] Wei Lu Mahbod Tavallaee, Ebrahim Bagheri and Ali A. Ghorbani. A detailed analysis of the KDD cup 99 data set. Proceedings of the 2009 IEEE Symposium Computational Intelligence for Security and Defense Applications, 47, July 2009.
[193] John McHugh. Testing intrusion detection systems: a critique of the 1998 and 1999 Darpa intrusion detection system evaluations as performed by Lincoln laboratory. ACM Transactions on Information and System Security, 3(4):262{294, 2000.
[194] Final Technical Memorandum. Benchmarks for evaluation of distributed denial of service Distribution, (January), 2008.
[195] Jelena Mirkovic, Erinc Arikan, Songjie Wei, Roshan Thomas, Sonia Fahmy, and Peter Reiher. Benchmarks for DDOS Defense Evaluation. Milcom 2006, (2):1{10, October 2006.
[196] Jelena Mirkovic and Peter Reiher. A taxonomy of DDoS attack and DDoS defense mechanisms. SIGCOMM Comput. Commun. Rev., 34:39{53, April 2004.
[197] George Nychis, Vyas Sekar, David G Andersen, Hyong Kim, and Hui Zhang. An empirical evaluation of entropy-based traffic anomaly detection. Proceedings of the 8th ACM SIGCOMM conference on Internet measurement conference IMC 08, page 151, 2008.
[198] Knuuti O. Intrusion detection system comparison in large ip-networks. Master's thesis, Tampere University of Technology, 2009.
[199] K Pawlikowski, H D J Jeong, and J S R Lee. On credibility of simulation studies of telecommunication networks. IEEE Communications Magazine, 40(1):132{139, 2002.
[200] S. Schwab, B. Wilson, and R. Thomas. Methodologies and metrics for the testing and analysis of distributed denial of service attacks and defenses. In Military Communications Conference, 2005. MILCOM 2005. IEEE, pages 2686-2692 Vol. 5, Oct. 2005.
[201] Joel Sommers, Hyungsuk Kim, and Paul Barford. Harpoon: a low-level traffic generator for router and network tests, page 392. ACM, 2004.
[202] Brian White, Jay Lepreau, Leigh Stoller, Robert Ricci, Shashi Guruprasad, Mac Newbold, Mike Hibler, Chad Barb, and Abhijeet Joglekar. An integrated experimental environment for distributed systems and networks. ACM SIGOPS Operating Systems Review, 36(SI):255, 2002.
[203] From Wikipedia. Principal component analysis principal component analysis (2nd ed.), by i. t. jollie, New York : Springer-Verlag , 2002 , isbn 0-387-95442-2 , xxix + 487 pp., 89.95 . Technometrics, 45(3):276-276, 2003.
[204] T. Ye, D. Veitch, G. Iannaccone, and S. Bhattacharya. Divide and conquer: Pc-based packet trace replay at oc-48 speeds. In Testbeds and Research Infrastructures for the Development of Net-works and Communities, 2005. Tridentcom 2005. First International Conference on, pages 262 - 271, Feb. 2005.
[205] Jaehak Yu, Hansung Lee, Myung-sup Kim, and Daihee Park. Traffic coding attack detection with snmp mib using svm. Computer Communications, 31(17):4212-4219, 2008.
[206] Kabiri P Zargar G R. Identication of effective network features for probing attack detection. 2009 1st International Conference on Networked Digital Technologies NDT 2009, pages 392-397, 2009.