

# ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΜΑΚΕΔΟΝΙΑΣ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ  
ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ



ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΔΥΤΙΚΗΣ ΜΑΚΕΔΟΝΙΑΣ

## ΕΛΕΓΧΟΣ ΑΣΦΑΛΕΙΑΣ ΚΑΙ ΑΚΕΡΑΙΟΤΗΤΑΣ ΣΕ ΔΙΚΤΥΑ ΚΡΙΣΙΜΩΝ ΥΠΟΔΟΜΩΝ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Δημήτριος Ιωσηφίδης (ece01336)

Επιβλέποντες Καθηγητές:

Αναπληρωτής καθηγητής Παναγιώτης Σαρηγιαννίδης

Επίκουρος καθηγητής Αλέξανδρος-Απόστολος Μπουλογεώργιος

Σεπτέμβριος 2022, Κοζάνη

**UNIVERSITY OF WESTERN MACEDONIA**

DEPARTMENT OF ELECTRICAL AND COMPUTER  
ENGINEERING



UNIVERSITY OF  
WESTERN MACEDONIA

**PENETRATION TESTING IN CRITICAL  
INFRASTRUCTURES**

DIPLOMA THESIS

Dimitrios Iosifidis (ece01336)

Supervisors:

Associate Professor Panagiotis Sarigiannidis

Assistant Professor Alexandros-Apostolos Mpoulogeorgios

September 2022, Kozani

# Περίληψη

Στην συγκεκριμένη διπλωματική έγινε η ανάλυση της χρησιμότητας των κρίσιμων υποδομών στην σημερινή εποχή, της κυβερνοασφάλειας και αναφερθήκαμε πιο συγκεκριμένα στην ασφάλεια των κρίσιμων υποδομών και το πόσο μπορεί αυτή να επηρεάσει την καθημερινότητα των πολιτών.

Αναφερθήκαμε επίσης στην χρησιμότητα του ελέγχου ασφάλειας και ακεραιότητας (penetration testing) καθώς και στον τρόπο με τον οποίο διεξάγεται ένας κύκλος ελέγχου όπου αποτελείται από τα 6 στάδια. Δόθηκαν επίσης κάποια εργαλεία που βοηθάνε στην διεξαγωγή πληροφοριών σε ένα σύστημα για την εύρεση ευπαθειών.

Στην συνέχεια έγινε αναφορά στα δύο δίκτυα/υποδομές τα οποία μελετήθηκαν ώστε να διεξαχθεί η συγκεκριμένη διπλωματική τα οποία ήταν ένα Software-Defined Network Controller(SDN Controller) και ένα σύστημα υποδομών φόρτισης για ηλεκτρικά οχήματα (Charging Infrastructures for Electrical Vehicles).

Καθόλη την διάρκεια της εύρεσης ευπαθειών στα δυο αυτά συστήματα χρησιμοποιήθηκαν τεχνικές και εργαλεία Penetration testing όπως nmap, nessus, docker εντολές κτλ.

Τέλος παρουσιάζονται τα αποτελέσματα που εξήχθησαν από την διαδικασία του penetration testing και γίνεται αναφορά σε όλες τις ανοιχτές πόρτες (ports) που υπήρχαν σε κάθε δίκτυο καθώς και τις εντολές που χρειάστηκαν ώστε να πάρουμε πρόσβαση.

## Abstract

In this thesis we analyzed the usefulness of critical infrastructure in today's era of cybersecurity and we referred more specifically to the security of critical infrastructure and how it can affect the daily lives of citizens.

We also discussed the importance of penetration testing and how to conduct a 6-step audit cycle, as well as some tools that help to conduct information on a system to find vulnerabilities.

Then, a reference was presented to the two networks/infrastructures that were studied to carry out this thesis which were a Software-Defined Network Controller (SDN Controller) and a Charging Infrastructures for Electrical Vehicles system.

Throughout the process of finding vulnerabilities in these two systems, Penetration testing techniques and tools such as nmap, nessus, docker commands etc. were used.

Finally, the results obtained from the penetration testing process are presented, along with the open ports that existed in each network and the commands needed to gain access.

# Keywords

Critical Infrastructure, Cyber Security, Penetration Testing, SDN Controller, Charging Infrastructure.

## Table of Content

Περίληψη	3
Abstract	3
<b>1. Introduction</b>	<b>9</b>
<i>1.1 Security in Critical Infrastructures</i>	9
<i>1.2 Subject of Thesis</i>	10
<i>1.3 Thesis Organization</i>	11
<b>2. Related Work</b>	<b>11</b>
<i>2.1 Cyber Security of Critical Infrastructures</i>	11
<i>2.2 Reducing security vulnerabilities for critical infrastructure</i>	12
<i>2.3 Multi-Vendor Penetration Testing in the Advanced Metering Infrastructure</i>	13
<i>2.4 Tools for Penetration Testing in Critical Infrastructure</i>	14
<i>2.5 DIDEROT: An Intrusion Detection and Prevention System for DNP3-based SCADA Systems</i>	17
<i>2.6 Critical infrastructure protection system design based on SCOUT multitech security system for interconnected space control ground stations</i>	18
<b>3. Background</b>	<b>21</b>
<i>3.1 Penetration testing</i>	21
<i>3.2 Penetration Testing Boxes</i>	25
<i>3.2.1 White Box</i>	25
<i>3.2.2 Black Box</i>	26
<i>3.2.3 Grey Box</i>	26
<i>3.3 Penetration Testing Tools</i>	27
<b>4. Critical Infrastructures</b>	<b>44</b>
<i>4.1 Software-Defined Network/SDN controller</i>	44
<i>4.2 Charging Infrastructures for Electrical Vehicles</i>	45
<b>5. Results of the analysis of Critical Infrastructures</b>	<b>47</b>
<i>5.1 Analyzing the SDN Controller</i>	47
<i>5.1.1 Reconnaissance Phase of SDN Controller</i>	47
<i>5.1.2 Vulnerability Identification of SDN Controller</i>	48
<i>5.1.3 Exploitation Phase of SDN Controller</i>	52
<i>5.2 Analyzing the Charging Infrastructure for Electrical Vehicles</i>	54
<i>5.2.1 Reconnaissance Phase of Charging Infrastructure for Electrical Vehicles</i>	54
<i>5.2.2 Vulnerability Identification Phase of the Charging Infrastructure</i>	55
<i>5.2.3 Exploitation Phase of the Charging Infrastructure</i>	57
<b>6. Results</b>	<b>57</b>

<b>7. Conclusion</b>	58
<b>8. Future Work</b>	58
<b>9. Bibliography</b>	60

## List of tables

Table 1 – Tools for Penetration Testing .....	43
Table 2 - Nessus Results of SDN Controller .....	50
Table 3 - Nessus Results of Charging Infrastructure .....	56

## List of Images

Figure 1 - Control Loop.....	13
Figure 2 - National Infrastructure Protection Plan Risk Management Framework .....	15
Figure 3 - Critical Infrastructure Protection according to Risk Management.....	16
Figure 4 - Evaluation Results of the DIDEROT Detection Layer - Anomaly Detection.....	18
Figure 5 - SCOUT SENSNET general architecture.....	19
Figure 6 - Recover Architecture .....	20
Figure 7 - The Six Phases of Penetration Testing.....	22
Figure 8 - Risk Rating Scale.....	25
Figure 9 - Boxes of Penetration Testing .....	27
Figure 10 - Metasploit Dashboard .....	28
Figure 11 - w3af Dashboard .....	29
Figure 12 - Nessus Dashboard.....	30
Figure 13 - AppScan Dashboard.....	31
Figure 14 - Burp-Suite Dashboard.....	32
Figure 15 - Kali Linux GUI.....	33
Figure 16 - John The Ripper Options .....	34
Figure 17 - Hydra Options.....	35
Figure 18 – SQL-Map Options .....	36
Figure 19 - Nmap Scan.....	37
Figure 20 - Wireshark Dashboard.....	38
Figure 21 - Netsparker Dashboard .....	39
Figure 22 - Zed Attack Proxy Dashboard.....	40
Figure 23 - Nexpose Dashboard .....	41
Figure 24 - Core Impact Dashboard .....	42
Figure 25 - Results of the Nmap in SDN Controller.....	47
Figure 26 - Information about the Docker .....	53
Figure 27 - Information about the Docker .....	53
Figure 28 - Privilege Escalation on the Docker .....	54
Figure 29 - Nmap results of the Charging Infrastructure.....	55

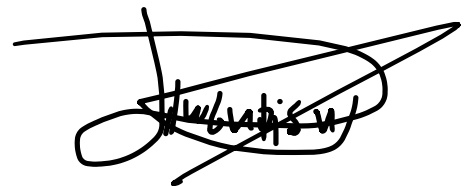
## **Δήλωση Πνευματικών Δικαιωμάτων**

Δηλώνω ρητά ότι, σύμφωνα με το άρθρο 8 του Ν. 1599/1986 και τα άρθρα 2,4,6 παρ. 3 του Ν. 1256/1982, η παρούσα Διπλωματική Εργασία με τίτλο “Έλεγχος ασφάλειας και ακεραιότητας σε δίκτυα κρίσιμων υποδομών” καθώς και τα ηλεκτρονικά αρχεία και πηγαίοι κώδικες που αναπτύχθηκαν ή τροποποιήθηκαν στα πλαίσια αυτής της εργασίας και αναφέρονται ρητώς μέσα στο κείμενο που συνοδεύουν, και η οποία έχει εκπονηθεί στο Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Πανεπιστημίου Δυτικής Μακεδονίας, υπό την επίβλεψη του μέλους του Τμήματος κ. Παναγιώτη Σαρηγιαννίδη και Αλέξανδρου-Απόστολου Μπουλογεώργιου και αποτελεί αποκλειστικά προϊόν προσωπικής εργασίας και δεν προσβάλλει κάθε μορφής πνευματικά δικαιώματα τρίτων και δεν είναι προϊόν μερικής ή ολικής αντιγραφής, οι πηγές δε που χρησιμοποιήθηκαν περιορίζονται στις βιβλιογραφικές αναφορές και μόνον. Τα σημεία όπου έχω χρησιμοποιήσει ιδέες, κείμενο, αρχεία ή / και πηγές άλλων συγγραφέων, αναφέρονται ευδιάκριτα στο κείμενο με την κατάλληλη παραπομπή και η σχετική αναφορά περιλαμβάνεται στο τμήμα των βιβλιογραφικών αναφορών με πλήρη περιγραφή.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα. Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και μόνο.

Copyright (C) Ιωσηφίδης Δημήτριος, Σαρηγιαννίδης Παναγιώτης και Μπουλογεώργιος Αλέξανδρος-Απόστολος, 2022, Κοζάνη

**Υπογραφή Φοιτητή:**





# 1. Introduction

## 1.1 *Security in Critical Infrastructures*

From a long time ago, the whole world has relied on things that provided for their daily needs, such as drinking water, food, transportation, and electricity. As the world evolved and the internet technology has become an integral part of peoples' life, needs such as telecommunications and the security of personal data arose. All these needs are called Critical Infrastructures (CI), and the world relies on them. We can think of Critical Infrastructure as the essential infrastructure for functioning societies. More specifically, Critical Infrastructures consist of all the facilities, transportation, services, and global economy of a nation.

More specifically to Critical Infrastructure, 16 sectors are considered critical and some of them are [1]:

- Critical Manufacturing Sector
- Information Technology Sector
- Emergency Services Sector
- Government Facilities Sector
- Healthcare and Public Health Sector
- Defense Industrial Base Sector
- Dams Sector
- Financial Services Sector
- Energy Sector
- Food and Agriculture Sector
- Communications Sector
- Nuclear Reactors, Materials, and Waste Sector
- Water and Wastewater Systems Sector
- Commercial Facilities Sector
- Transportation Systems Sector
- Chemical Sector

Such infrastructures must be protected. Events and incidents such as natural disasters, equipment failure, terrorist activities and cyber threats could jeopardise the protection of CI and cause devastating consequences. For instance, they might affect the operation of services such as the citizens' health care system, the transportation system, the banking sector and potentially threaten entire communities [2]. The more the population increase in the global community, the more critical it is to protect these infrastructures. While each country continuously tries to protect all these infrastructures, many reported incidents are in the opposite direction.

A recent event that has disturbed several states and affected people's daily lives and the infrastructure of several countries, is the pandemic caused by the COVID-19 [3]. COVID-19 broke out in November 2019, and since then, several countries have begun quarantine, while

others have prohibited entry and exit from their borders. This situation has had an impact on their economy as much as on the health-care system. Businesses have suspended the production and in addition, Covid-19 put pressure on health-care systems to treat citizens in order to prevent the virus from spreading [4].

Another attack that is worth noting is the attack that happened on the National Health Service (NHS). The “WannaCry attack” occurred on 12 May 2017. This virus encrypts data from infected computers and demands an exchange of a ransom payment to allow users access. When the attack happened, numerous appointments and surgeries were cancelled, forcing many patients to be transferred to different accident and emergency departments. This specific attack affected at least 81 out of 236 organizations across England [5]. As a result, this virus was one of the largest cyberattacks that affect the UK’s NHS and cost them a payment of £92M [6].

Except for natural disasters, several malicious threats must be addressed while defending CIs. Since the evolution of the smart systems, the interconnection between systems and the establishment of the Internet of Things (IoT) sector, the systems have become more vulnerable, providing an expanded attack surface allowing for sophisticated cyber-attacks, malicious actions and terrorist activities.

Therefore, it is crucial to protect to the greatest extent our CIs in order to have a smooth operation without affecting the people’s daily life. Ways to achieve this goal are [7] [8] [9]:

- Risk Management for Critical Infrastructure
- Improving Critical Infrastructure Security
- Improving Critical Infrastructure Resilience

Risk management will help analyze a problem and prioritize it depending on the risk, after which it will implement risk-mitigation measures to deal with it [10].

Consequently, improving physical security and cyber security is essential to protect CIs. That can happen by installing layers of security fences to protect important buildings and areas but also in the cyber field, companies must develop effective solutions to protect organizations’ networks, systems, and users, thus identifying and addressing virtual vulnerabilities.

Lastly, Critical Infrastructure must be resilient to changing conditions and capable of withstanding and recovering from disruptions. This involves the ability to protect against both physical and cyber-attacks, which requires implementing a cyber security defense program.

## ***1.2 Subject of Thesis***

This thesis focuses on Critical Infrastructures and especially on how to identify and overcome various vulnerabilities in these infrastructures. The methods that will be used are Data Analysis, Penetration Testing, and various other tools such as Metasploit, Nessus and Burp-Suite which will contribute to analyzing and securing these infrastructures. The systems that we will examine are:

- an SDN controller and
- a Charging Infrastructure for Electrical Vehicles.

The Data Analysis process will help in gathering information, organizing them, and drawing conclusions and insights that can help in making better decisions and preventing future attacks. Penetration testing will help in identifying and preventing security risks and the tools will assist in this operation by making the process easier and more functional.

### ***1.3 Thesis Organization***

In this section, we will provide you with a glimpse of the topics we will discuss. Firstly, we will provide background information on CI security and penetration testing. After, we will examine the approaches and strategies that we will employ, as well as the penetration testing tools. As a result, if feasible, we will solve any difficulties that may occur, and finally, we will evaluate all the programs and algorithms that we will employ.

## **2. Related Work**

In the field of Critical Infrastructure, there is a wide variety of work where some focus on achieving greater CI security, others analyse the data that already exist and others try to find vulnerabilities that may not have been identified yet. Below we present some respected papers related to CIs and the conclusions of these papers.

### ***2.1 Cyber Security of Critical Infrastructures***

A paper that focuses mainly on the security of Critical Infrastructures is "Cyber security of Critical Infrastructures" [11]. In this paper the SCADA system is utilized to tackle numerous cyber threats.

The SCADA system is a critical tool for some businesses as it helps maintain efficiency, process data to make smart decisions, and can communicate between systems to solve problems more quickly so there is less downtime [12].

Behind the operation of SCADA systems, two systems can operate. The first is the Programmable Logic Controller or PLC and the second one is the Remote Terminal Units or Remote Telemetry Units where RTU is written in short.

PLC and RTU are electronic devices that we use in SCADA systems. More specifically, PLCs are digital computers that are mainly used for the automation of electromechanical processes. They are specifically intended for output configurations with many inputs.

On the other side, the RTU is being controlled by microprocessors and its main function is to interface SCADA with objects where they have "physical value".

The main difference between RTU and PLC is that RTUs are more convenient because of the wireless communication which can be used over larger geographical areas. PLC, on the other hand, is better utilized in local controls such as production lines or plants [13] & [14].

An attack that affected and aroused the attention of many individuals was STUXNET which it was a computer virus particularly intended for targeting Windows-based industrial computers and gaining control of Programmable Logic Controllers. That attack was able to alter the behavior of remote actuators, resulting instability on the system.

Because SCADA is useful for connecting IT networks in order to provide better and quicker communications, the security of SCADA communications is getting more challenging.

Knowing the great evolution of SCADA systems, solving various new cyber threats can become easier as more methods can be applied to such systems. Finally, it emphasizes the fact that such systems in the future will be able to focus on larger problems in real-time and have a higher accuracy of results.

## ***2.2 Reducing security vulnerabilities for critical infrastructure***

Another very apt paper on reducing vulnerabilities in the critical infrastructure sector is "Reducing Security Vulnerabilities for critical infrastructure". This paper provides an outline of the importance of improving the Process Control System(PCS) [15].

Process Control is the ability to regulate and adjust a process in order to get the desired results. Its utility in an organization is to maintain quality by improving performance [16].

To keep all process running properly and also with the desired level of quality, the Control Loop is used. Every decision in this loop is made in cooperation with the rest of the system. After locating the adjustment point, the control loop initiates a four-step process.

The four steps are [17]:

- Sense
- Compare
- Respond
- Affect

The procedure is as follows: The *Sensor* identifies where the process is located using a sensor. This sensor can be either thermocouples or RTDs. Next, we use an electronic PID selector to evaluate the current state relative to the setpoint. This is the *Compare* stage. After this process is done, we check the *Response* which informs us in case of an error between the temperature value and the temperature set point. Finally, in the *Affect* step, the process variable changes if needed. This loop is executed repeatedly and affects the whole process.

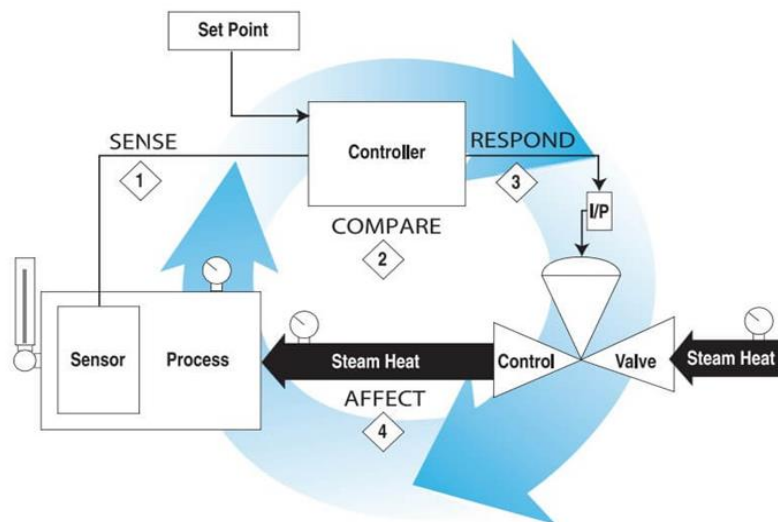


Figure 1 - Control Loop<sup>1</sup>

Though there are some suggestions to avoid cyber threats such as the research and improvement of SCADA systems by developing encryption algorithms. Afterwards, proposes the use of authentication and risk analysis for the control system and encrypted communication protocols. Lastly, focuses on studying safety methods at national security policy because many companies such as those dealing with water and wastewater facilities, electric generation, satellites, oil and gas facilities, etc. rely heavily on SCADA systems.

### 2.3 Multi-Vendor Penetration Testing in the Advanced Metering Infrastructure

One piece of Critical infrastructure is energy and electrical grids. In the paper "Multi-vendor Penetration Testing in the Advanced Metering Infrastructure" we will look at Advanced Metering Infrastructure (AMI) and how it is rapidly changing the evolution of electrical grids [18].

Electrical grids are circuits where electricity is transferred in Alternative Current (AC) from generators to consumers. There are different sizes of electrical grids where they are categorized into [19]:

- Power stations
- Electrical substations
- Electrical power transmission
- Electric power distribution

More specifically for AMI, it consists of a two-way communication system that allows communication between utilities and customers. One of the features that AMI provides is the ability to capture and filter data so that it can be accessed at any time. It is also possible to

<sup>1</sup> Control Loop <<https://instrumentationtools.com/what-is-control-loop/>>

monitor the termination points so that in case of any voltage disturbance there can be an easy real-time response. The data that have been captured are updated every 15 minutes [20].

The paper then focuses on penetration testing and its approach, the attack tree. In fact, an attack tree is the way that we enumerate the attacks in order for the targeted attack to succeed. This can be accomplished by splitting down the attack into smaller sub-attacks, making the procedure easier. However, in the set of sub-attacks we achieve the initial attack. This process has both positive and negative features.

Initially, this attack tree can work for most attacks on smart metering systems. However, the downside is that each leaf of the tree does not go into details about the system. As a result, it cannot always find all the vulnerabilities.

To be able to fix this, they retrieved the tree and broke it into 2 types of attack trees:

- An Archetypal Tree
- and Concrete Tree

The archetypal tree adopts a specific architecture and enumerates the attacks in order to reach the final attack, whereas the Concrete tree refines the attack using information from the vendor's system.

More generally, the methodology is as follows:

1. Capture architectural description
2. Construct archetypal tree
3. Capture vendor-specific description
4. Construct concrete trees
5. Perform Penetration Testing

The paper also studied a strategy for evaluating the security of several devices deployed in the AMI. This effort is intended not just to speed security analysis, but also to assure broader and more consistent coverage of potential attacker intentions and techniques. In future work they will extend their work to other vendor devices, as well as expand the foundation of attacker goals and associated trees.

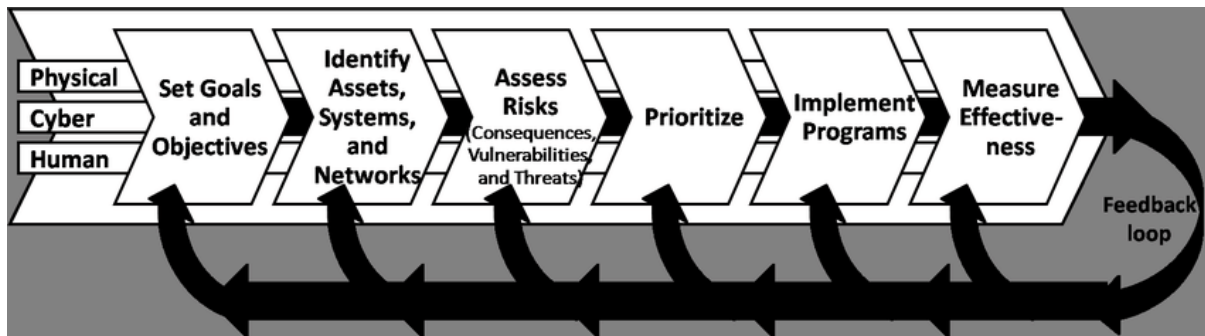
## ***2.4 Tools for Penetration Testing in Critical Infrastructure***

Apart from those papers, some tools help to identify threats and analyze them. The Critical Infrastructure Protection tools can be categorized as [21] [22]:

- The way they operate (Functionality) [23].
- Technical Modeling Approach.

More specifically in the functionality group, we have the following subcategories, where according to the National Infrastructure Protection Plan (NIPP), some goals for the security risks can be achieved in the order of:

1. Risk Prioritization (RP),
2. Risk Identification (RI),
3. Risk Mitigation Planning (RMP),
4. Risk Assessment (RIA),
5. Effectiveness Evaluation (EE).



*Figure 2 - National Infrastructure Protection Plan Risk Management Framework<sup>2</sup>*

While in the group with the technical modeling approach we have [24]:

1. Empirical Approaches: It is research-based on the observation of historical events or collapsed data. Some of the characteristics that empirical approaches have are that they can spot failure patterns, they can calculate metrics so to help in decision-making, and they can provide solutions to reduce risks that may occur in infrastructures.
2. System-Dynamics Based Approaches: It is an analyzer that provides methods and tools for complex adaptive systems.
3. Agent-Based Approaches: Emphasizes in learning through organism-environment interaction. due to the complexity of CI and the decision they have to make, CIs are based on Complex adaptive Systems. A complex adaptive system is a type of complicated system that can adapt to its environment. As with other complex systems, they are made up of numerous pieces, known as agents, who interact in a nonlinear manner to form a network of links in which agents act and respond to one another's activity.
4. Network-Based Approaches: In CIs different components are represented via nodes. Additionally, the physical and relational ties between them are represented by links. Another characteristic is that they have the capability to represent topologies, as long as flow patterns.
5. Other Approaches: There are more than these approaches to analyze CIs. Approaches include mathematical equations, economical interdependencies, and others.

<sup>2</sup>National Infrastructure Protection Plan Risk Management Framework  
[https://www.researchgate.net/figure/US-NIPP-risk-management-framework\\_fig2\\_229067287](https://www.researchgate.net/figure/US-NIPP-risk-management-framework_fig2_229067287)

In general, there are more than 68 tools for risk management. Apart from that, a great number of the tools start with the risk identification stage and only few of them face the first two stages, while even fewer can be managed by the Risk Management Framework.

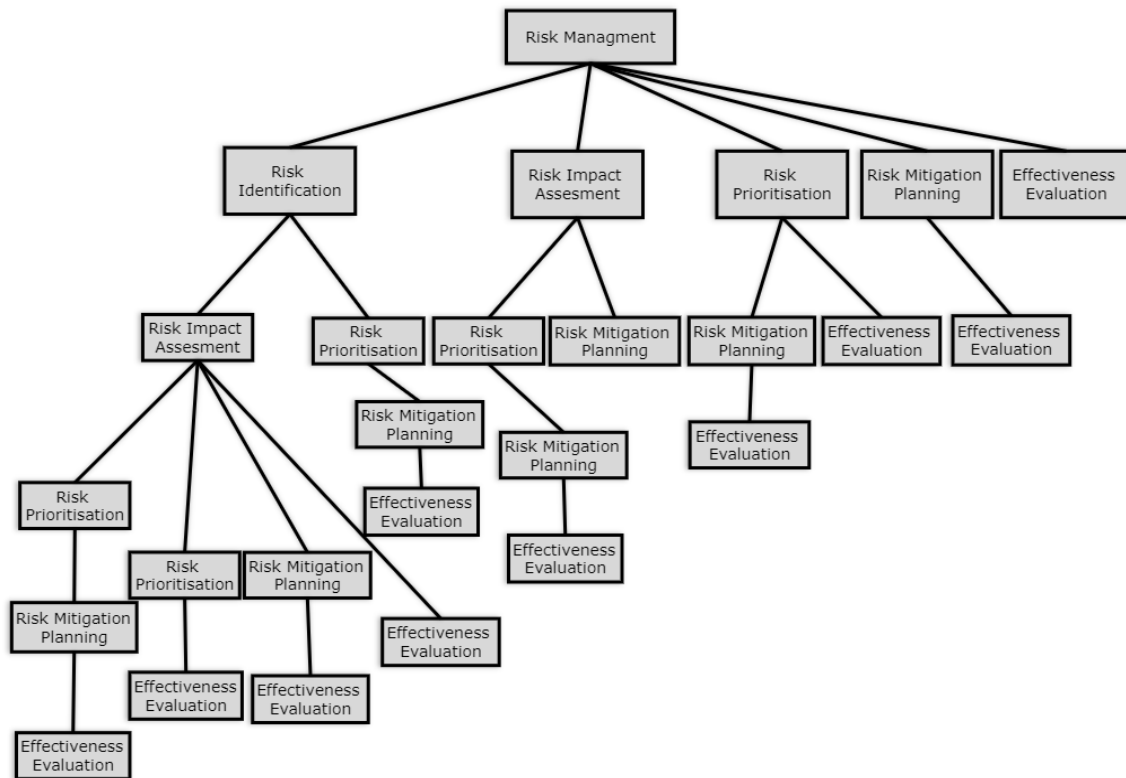


Figure 3 - Critical Infrastructure Protection according to Risk Management<sup>3</sup>

The CI modeling and analysis tools, frameworks, and approaches give their attention to studies of large-scale infrastructures and their states. The majority of Critical Infrastructure Protection strategies are built on risk management frameworks.

Accordingly, the USA/NIPP maintains the most sophisticated program for developing strategies, methodologies, and applications. We can categorize the classification into two stages:

- The first part uses purpose stages to gain a better understanding of the goals achieved and the functioning of each categorized CI tool.
- The second part of this classification categorized tools based on their technical modeling methodologies.

The degree to which government and business sector partners engage in systematic, effective, multi-directional information exchange determines the effectiveness of CIP strategies.

<sup>3</sup> . Critical Infrastructure Protection according to Risk Management  
 <<https://infosec.aueb.gr/Publications/CIP-2016%20CIP%20Tools.pdf>>



Future study and tools should target the latter phases of the Risk Assessment framework and either focus more on establishing comprehensive methods that can model all CI sectors or focus solely on particular challenges in specific sectors.

## ***2.5 DIDEROT: An Intrusion Detection and Prevention System for DNP3-based SCADA Systems***

As mentioned before, SCADA system play an important role in Critical Infrastructures because it can identify malicious network traffic [25]. A pioneering paper that involves SCADA systems is "DIDEROT". DIDEROT actually stands for, Dnp3 Intrusion Detection Prevention System, which is based on 2 functions, supervised Machine Learning as well as unsupervised Machine Learning [26].

To be explicit, supervised learning implies that the algorithm is performed with some data and some examples of how that model will be executed and trained.

On the other hand, unsupervised learning learns a pattern from an untagged data set.

Now more specifically, in this paper they created DIDEROT which is based on the network flow of the network and can recognize 2 detection layers:

- Intrusion detection
- Anomaly detection

The way DIDEROT works is divided into two stages.

In the first stage, it starts the supervised Machine Learning method and looks for any cyberattack on DNP3. The search for any anomaly in DNP3 includes the following:

- Injection
- Flooding
- DNP3 reconnaissance
- Replay attacks
- Masquerading.

After passing the first stage and not finding any kind of unusual traffic in the network, the second stage begins, in which unsupervised ML is used to detect any anomalies. In case it is not detected then the network traffic proceeds normally.

In general, DIDEROT has the following features:

- Detection of DNP3 cyberattacks
- Detection of DNP3 anomalies
- Evaluating Machine Learning methods so to recognize any anomalies or cyberattacks
- Mitigating DNP3 cyberattacks/anomalies

And it consists of 3 modules:

- DIDEROT Analysis Engine
- Response Module
- Data monitoring module

Additionally, the data monitoring gives the DIDEROT analysis engine data to detect anomalies or cyberattacks. Then the DIDEROT analysis engine is trying to detect any anomaly or cyberattack with the help of Intrusion Detection and Anomaly Detection. Lastly, the response module generates security events if needed.

Finally, the results of this paper were quite ambitious. As shown in the table below, DIDEROT had an accuracy of 0.951 which is 95%. Moreover, a future plan is presented to create several rules that will help to identify new DNP3 anomalies.

<b>ML Method</b>	<b>Accuracy</b>	<b>TPR</b>	<b>FPR</b>	<b>F1</b>
MCD	0.946	1	0.107	0.949
LOF	0.942	1	0.114	0.945
PCA	0.5	0	0	0
Isolation Forest	0.950	1	0.098	0.953
<b>DIDEROT Autoencoder</b>	<b>0.951</b>	<b>1</b>	<b>0.097</b>	<b>0.953</b>

Figure 4 - Evaluation Results of the DIDEROT Detection Layer - Anomaly Detection<sup>4</sup>

## ***2.6 Critical infrastructure protection system design based on SCOUT multitech security system for interconnected space control ground stations***

The paper "Critical infrastructure protection system design based on SCOUT" [2] uses the SCOUT Multitech security system to detect various attacks and anomalies in satellite links and/or even to protect the space control ground station. SCOUT is a proof-of-concept system based on 3 categories [27].

- CYBERSENS
- SENSNET
- RECOVERY

Now, each individual category offers something different. More specifically, CYBERSENS has 4 phases where it performs.

<sup>4</sup> Evaluation Results of the DIDEROT Detection Layer - Anomaly Detection  
[https://www.researchgate.net/publication/343853580\\_DIDEROT\\_an\\_intrusion\\_detection\\_and\\_prevention\\_system\\_for\\_DNP3-based\\_SCADA\\_systems](https://www.researchgate.net/publication/343853580_DIDEROT_an_intrusion_detection_and_prevention_system_for_DNP3-based_SCADA_systems)

1. Gathering data phase: in this phase, we collect data from the network and host probes to give them to the detection engine.
2. Amassing of data phase: in this phase, the data that we collect from the gathering phase is not feasible to process so we use some techniques to the data so to be ready for actual detection.
3. Detection phase: it finds any anomaly of malicious behaviour by applying anomaly methods in the system. As long as the technique works it protects the sensitive data that is collected from each probe.
4. Alert phase: as obvious as it is, this phase informs if anomalies are detected. If so, measures have to be taken and CYBERSENS informs the MCU by sending alerts about the detected incident.

For the second category, SENSNET has a local component named S-MCU which is connected to the sensor network.

The functionalities that S-MCU provides are to keep alive the communication with detection nodes as well as with the main MCU. Apart from that, can perform data fusion and pick the targets to display. (Fig. 5)

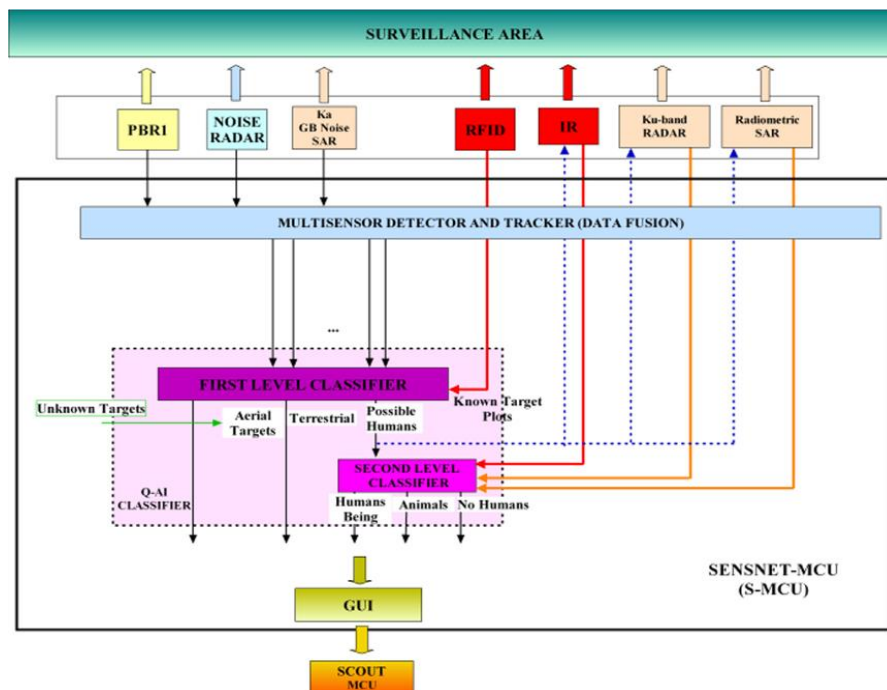


Figure 5 - SCOUT SENSNET general architecture<sup>5</sup>

The last category is the RECOVERY architecture. In this architecture, the SCOUT framework's RECOVER subsystem is designed with a centralized approach and adheres to the emerging

<sup>5</sup> SCOUT SENSNET general architecture

<<https://www.sciencedirect.com/science/article/pii/S1874548220300718>>

SDN. The main segment of the SCOUT architecture, as shown in Fig. 6, will be ruled by an SDN controller that implements a policy-based secure routing application. Additionally, the SDN will provide a traffic recovery if links or nodes crash.

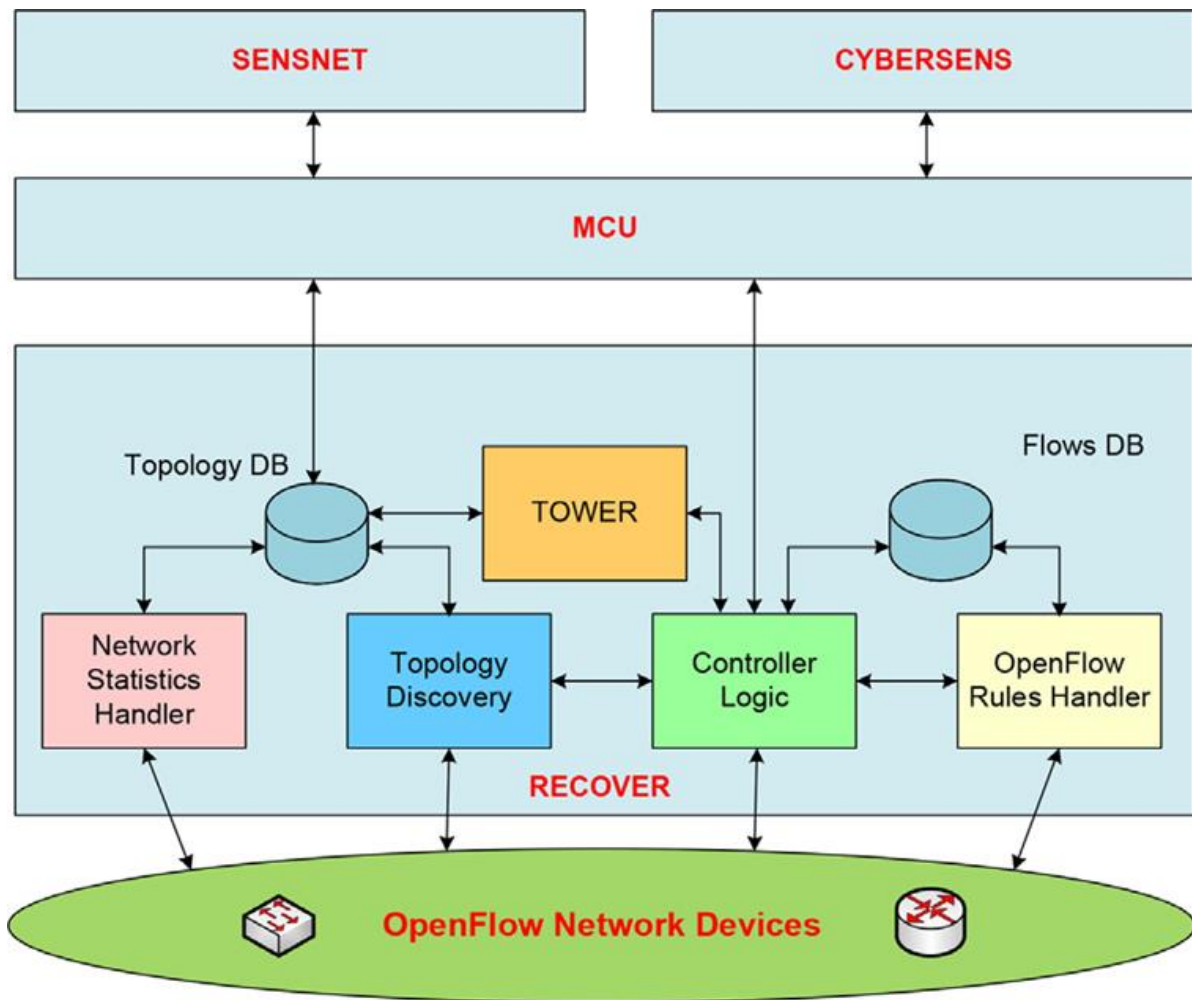


Figure 6 - Recover Architecture<sup>6</sup>

The goal of this paper was to test the SCOUT system on both cyber (CYBERSENS) and physical (SENSNET) thread detection and identification.

The SENSNET subsystem focuses on physical surveillance. This system is able to:

- work in perpetuity,
- identify people and vehicles that are authorized,
- distinguish people, animals, and other objects, as long as it offers anonymity to people,
- and provides low installation and maintenance cost.

<sup>6</sup> RECOVER architecture <<https://www.sciencedirect.com/science/article/pii/S1874548220300718>>

The network's data was processed in two steps. First phase passes through the S-MCU, then it continues to the MCU to confirm that there is no anomaly. In case of an anomaly, the MCU creates a target classification in the S-MCU.

Afterwards, the CYBERSENS is used to protect critical infrastructures from cyber-attacks. It provides detection to the system while keeping alive the communication and maintaining the privacy of the critical infrastructure users. This subsystem runs advanced Intrusion Prevention/Detection Systems for critical infrastructures.

Lastly, the RECOVERY subsystem is based on an SDN model that uses the OpenFlow standard protocol. Now the recovery subsystem provides a traffic control application that identifies security routing and packet filtering by taking advantage of the SDN control platform.

This specific model of SCOUT is a demonstration of a reduced version of the complete SCOUT system that could be used in future work.

## 3. Background

It is a priority these days with all this evolution of technology to protect people from cyber attacks. Penetration Testing is both complex and a very fragile process. The job of a penetration tester is to test a client's system to find any and every vulnerability while, at the same time, guaranteeing that the activity will have the least impact possible on the production systems and services.

### 3.1 *Penetration testing*

The penetration testing process [28] is divided into six parts, the first of which is *Pre-Engagement Interaction* in which the company contacts the penetration testing team and makes an agreement about the way that they will work for the company. After that, the penetration testers start the process of *Gathering Information* about the system. Once all the necessary information needed by the team has been obtained, the *Threat Modeling and Vulnerability Identification* process begins. That means, they perform a “mapping” process on the network to identify any potential risks that the network may have. Subsequently in this phase, the team continues with the *Exploitation* process in which they try to exploit the vulnerability that they found. In the last two stages, which are *Post-Exploitation and Reporting*, the risk rate of each vulnerability is analyzed, and a report is created for the company.

The stages are depicted and discussed separately below:

- Pre-Engagement Interactions
- Reconnaissance or Open-Source Intelligence (OSINT) Gathering
- Threat Modeling & Vulnerability Identification
- Exploitation
- Post-Exploitation, Risk Analysis & Recommendations

- Reporting

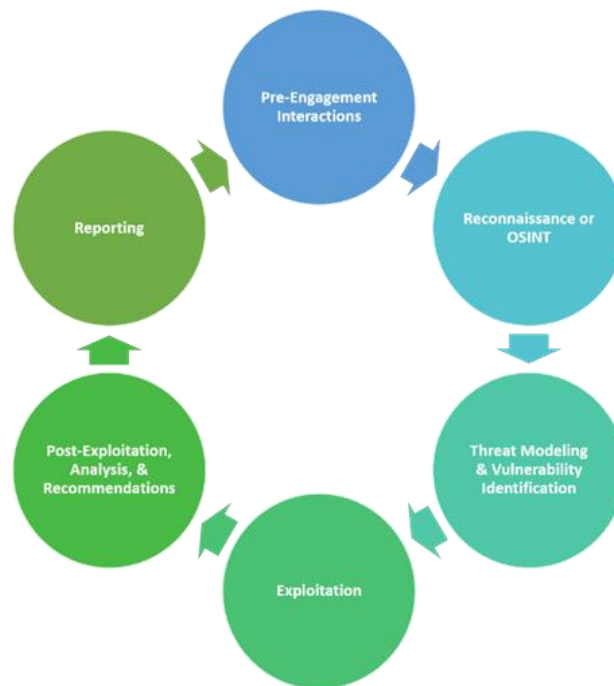


Figure 7 - The Six Phases of Penetration Testing<sup>7</sup>

### Pre-Engagement Interaction

In this phase, a penetration testing company will outline the logistics of the test, expectations, legal implications, objectives, and goals that the customer would like to achieve.

The penetration testers should work with the company to fully comprehend any risks, your organizational culture, and the best pen-testing strategy for your organization. There are different boxes that a penetration testing team can perform, and those boxes are white box, black box, or grey box.

### Reconnaissance or Open-Source Intelligence (OSINT) Gathering

Information gathering is an important first step in penetration testing. During this phase, a pen-tester gathers as much information about your company as they can and potential targets for exploitation [29].

Depending on which type of pen-test you agree upon (white-box, grey-box, or black-box), the penetration tester may have varying degrees of information about your organization or may need to identify critical information on their own to uncover vulnerabilities and entry points in your environment.

---

<sup>7</sup> The Six Phases of a Penetration Testing <<https://cipher.com/blog/a-complete-guide-to-the-phases-of-penetration-testing>>

## Threat Modeling & Vulnerability Identification

In the footprinting phase, the tester identifies targets and exploits that have access to attack. Any information gathered during the Reconnaissance phase is used to guide the penetration testing process.

The most common areas a pen-tester will search includes:

- Business assets - Items of value owned by a company
  - Customer data
  - Technical data
  - Employee data
- Threats – Either internal and/or external threats
  - Internal threats such as employees, management, vendors, etc.
  - External threats such as ports, Web Applications, Network Protocols and Traffic, etc.

In this phase, a pen-tester can use vulnerability scanner tools to scan the network and discover security risks. The next step is to validate if the vulnerability is exploitable. All the results are concluded in the reporting phase.

## Exploitation

After the mapping from the above step, the pen-tester starts the exploitation part within the network, applications, and data. The goal of penetration testers is to see how far they can go into the company's system, find high-value targets and avoid being detected.

In case the company puts boundaries in the testing phase then the tester will go as far as determined by the guidelines.

**Some of the standards exploit tactics include:**

- Network Attacks
- Physical Attacks
- Wi-Fi attacks
- Web Application Attacks
- Memory-based attacks
- Social engineering
- Zero-Day Angle

After the exploitation, the penetration tester will also review and make a document with all the vulnerabilities and how these vulnerabilities are exploited. Apart from that, they must refer to the techniques and tactics that they use to get access to the targets.

## Post-Exploitation, Risk Analysis & Recommendations

After the exploitation phase, we have Post-Exploitation, Risk Analysis & Recommendations phase. In which the goal of this phase is to document the company about the exploitations and the way the tester gains access to the valuable pieces of information.

The penetration tester should be able to determine the value of the compromised systems as well as the impact of any sensitive data gathered. On top of that, the penetration tester must give some recommendations on how to protect and fix those vulnerabilities. Once the company is informed about the recommendation the tester should clean up the environment.

A typical clean up includes [30]:

- Removing any user accounts created to connect to the compromised system
- Reconfiguring settings back to the original parameters
- Removing any files or scripts from the compromised systems
- Eliminating any rootkits installed in the environment

## Reporting

Last but not least, this phase is as important as the others. Reporting phase includes any recommendations from the penetration testing company and allows the company to review the report's conclusions with the ethical hackers [30].

In this report, you should be able to understand exactly how the exploitations were discovered as well as recommend ways to protect those security issues.

In this documentation, the penetration tester will value those vulnerabilities based on how critical it is for the company. The scale that follows shows the importance of the risk that has been exploited.



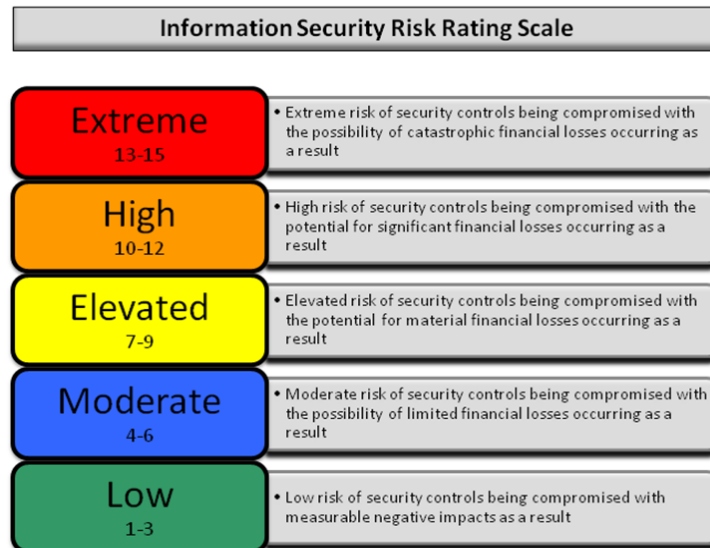


Figure 8 - Risk Rating Scale<sup>8</sup>

## 3.2 Penetration Testing Boxes

Now that we explained what penetration testing is and how we separate and value the risks of the vulnerabilities, we can categorize the boxes into three groups as we mentioned before.

Types of boxes [31]:

- White box
- Grey box
- Black box

The difference between these three boxes is how much information the penetration testers have before they start the process of scanning and finding vulnerabilities.

### 3.2.1 White Box

More specifically, the White box involves sharing all the information that they have for the network and system with the penetration testers. In this way, the tester saves time and reduces the overall cost of an engagement. A white box penetration testing simulates a targeted attack on a system using as many attack paths as feasible. A white box can be seen as a crystal box or oblique box penetration testing [32].

Advantages of the White box:

- In-depth testing.
- The penetration testers team saves time since they already know the details for the box.

<sup>8</sup> Risk Rating Scale <<https://cipher.com/blog/a-complete-guide-to-the-phases-of-penetration-testing/>>

- Testing any and every area including code and program flows.
- Cheaper than the black box.

Disadvantages of the White box:

- It is not as realistic as the black box.
- There is a different approach the pen-tester tests the machine because of the details that he gets.

### ***3.2.2 Black Box***

In the Black box, no information is provided to the tester at all. The tester must find a way to get access without any help from the company. This scenario can be seen as the most authentic because it demonstrates a real attack on the target system. However, because of the difficulty that box has and the lack of information that is given is the costliest option [32].

Some phases of the Black box:

- Remote access exploitation
- Server-level vulnerabilities
- Network scanning
- Social engineering

Advantages of the Black box:

- More realistic
- The tester investigates the machine as an attacker and not as a tester
- Allows for the detection of weak points in functional performance
- Because the tester works separately from the developers, makes the results more accurate in terms of their realism

Disadvantages of the Black box:

- Time-consuming because the tester must check every path.
- Some scenarios are difficult to test without any blueprint from the company.
- More expensive than the white box.

### ***3.2.3 Grey Box***

The last box is the Grey box of penetration testing, also known as a translucent box, is neither a white box with full information given nor a black box with no information. In this box, only limited information is shared with the tester. A Grey box testing can be performed by end-users and also by pen-testers and developers.

Advantages of the Grey box:

- Grey box testing includes the advantages of both white box and black box testing.
- The testing will be done from the perspective of the user rather than the designer.
- Due to the limited knowledge provided, grey-box testers can create excellent test scenarios focusing on communication protocols and data type handling.
- Grey box testers rely on interface definition and functional specifications instead of source code.

Disadvantages of the Grey box:

- Since access to the source code is not available. Going through the code and testing coverage is difficult.
- If the software designer has previously run a test scenario, the tests may be redundant.
- Testing every potential input stream would take an excessive amount of time, many software pathways will go untested.

### Differences between Types of Penetration Testing

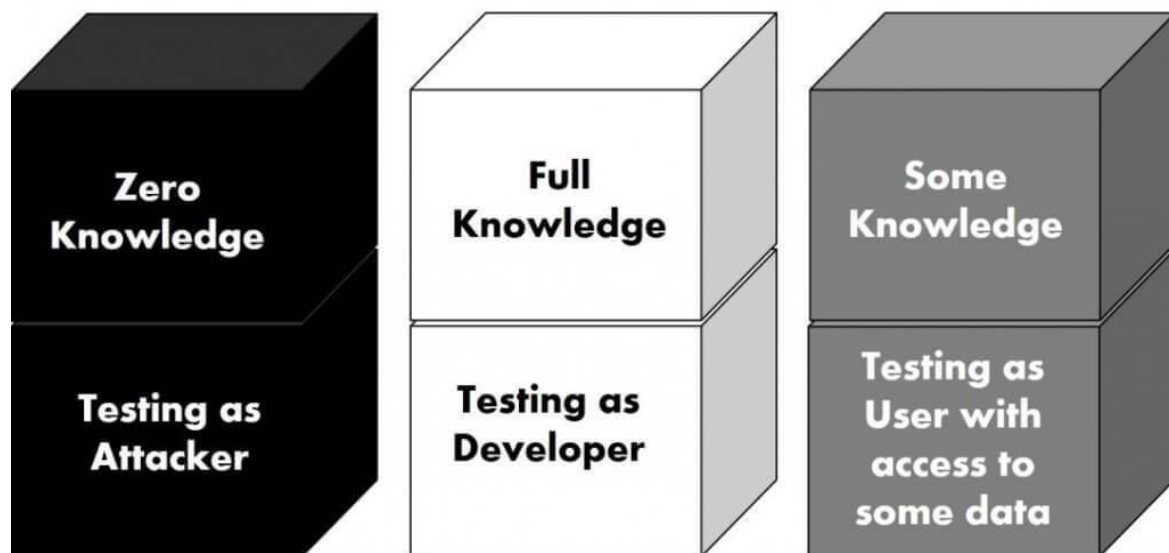


Figure 9 - Boxes of Penetration Testing<sup>9</sup>

### 3.3 Penetration Testing Tools

During the information gathering phase and after, penetration testers usually use some tools to dig out information either from the network or the system. Some of those tools are open source and others are premium. Those tools help the tester to make the process faster and more accurate [33]. Below we present some tools that a penetration tester can use.

---

<sup>9</sup> Boxes of Penetration Testing <<https://www.coresentinel.com/black-box-vs-white-box-testing/>>

# Metasploit

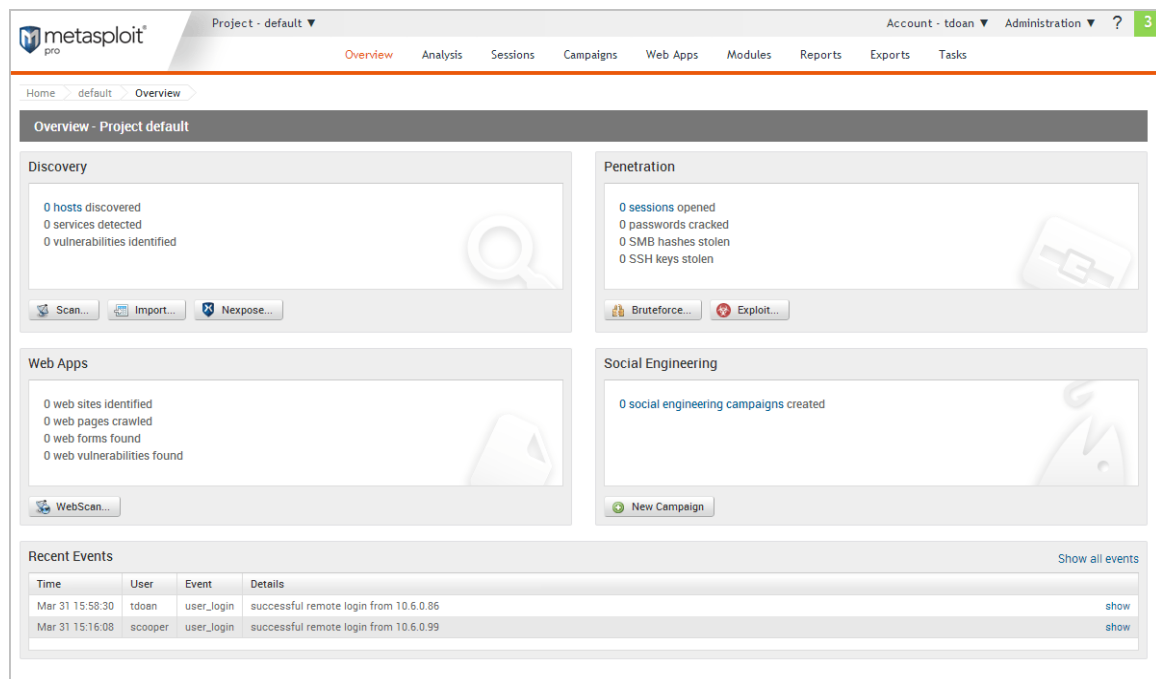


Figure 10 - Metasploit Dashboard<sup>10</sup>

The Metasploit framework is a very valuable tool that can be used by both cyber criminals and ethical hackers as well as penetration testers to investigate vulnerabilities on networks and servers. With this tool, the penetration testing team can deploy a ready-made code or even customize it and search for weaknesses [34].

Metasploit features:

- It offers both a command-line and a graphical user interface.
- It is compatible with Linux, Windows, and Mac OS X.
- The Metasploit community edition is free to the InfoSec community.
- Vulnerability scanner and network discovery.
- Basic exploitation.

<sup>10</sup> Metasploit Dashboard <[https://docs.rapid7.com/api/docs/file/product-documentation\\_\\_master/metasploit/images/ui-dashboard-overview.png](https://docs.rapid7.com/api/docs/file/product-documentation__master/metasploit/images/ui-dashboard-overview.png)>

## w3af

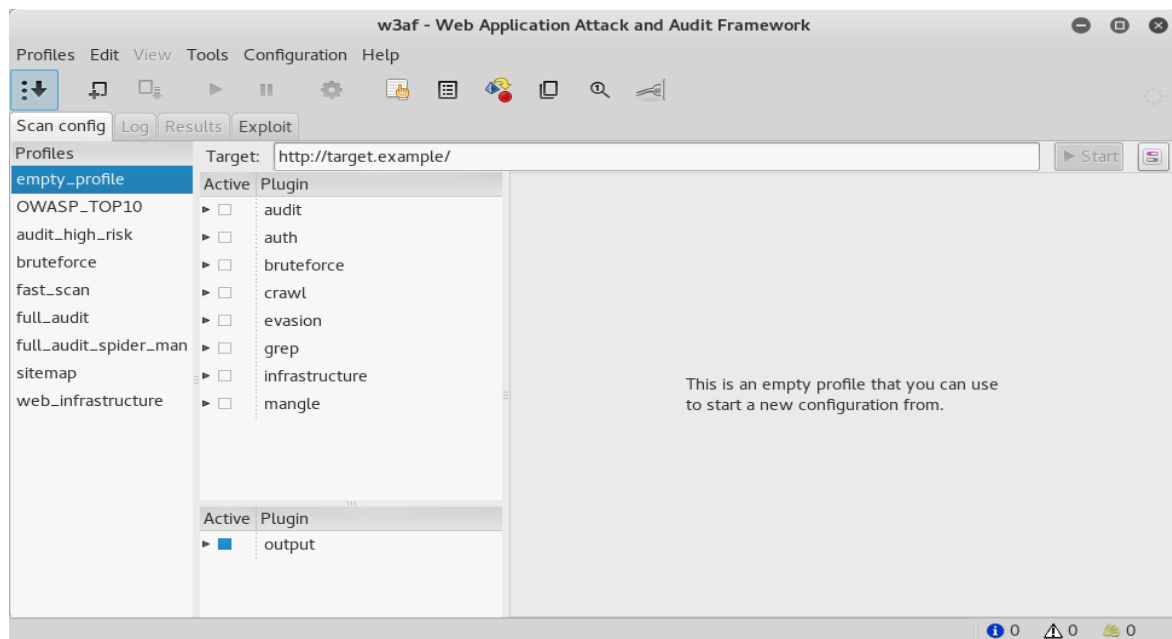


Figure 11 - w3af Dashboard<sup>11</sup>

W3af is an open-source tool that can be used for web application vulnerability scanners. With this tool, we exploit all web application vulnerabilities so to secure them afterward. It detects over 200 vulnerabilities such as Cross-Site Scripting (XSS), SQLi, PHP misconfigurations and lowers the site's overall risk [35].

w3af features:

- Proxy support
- Cookie handling
- HTTP response cache
- DNS cache
- File upload using multipart
- Add custom headers to requests
- UserAgent faking
- HTTP Basic and Digest authentication

---

<sup>11</sup> w3af GUI <<https://static.packt-cdn.com/products/9781783982165/graphics/95f1bf69-f29c-42ba-94b4-3eaaa28d30bd.png>>

## Nessus

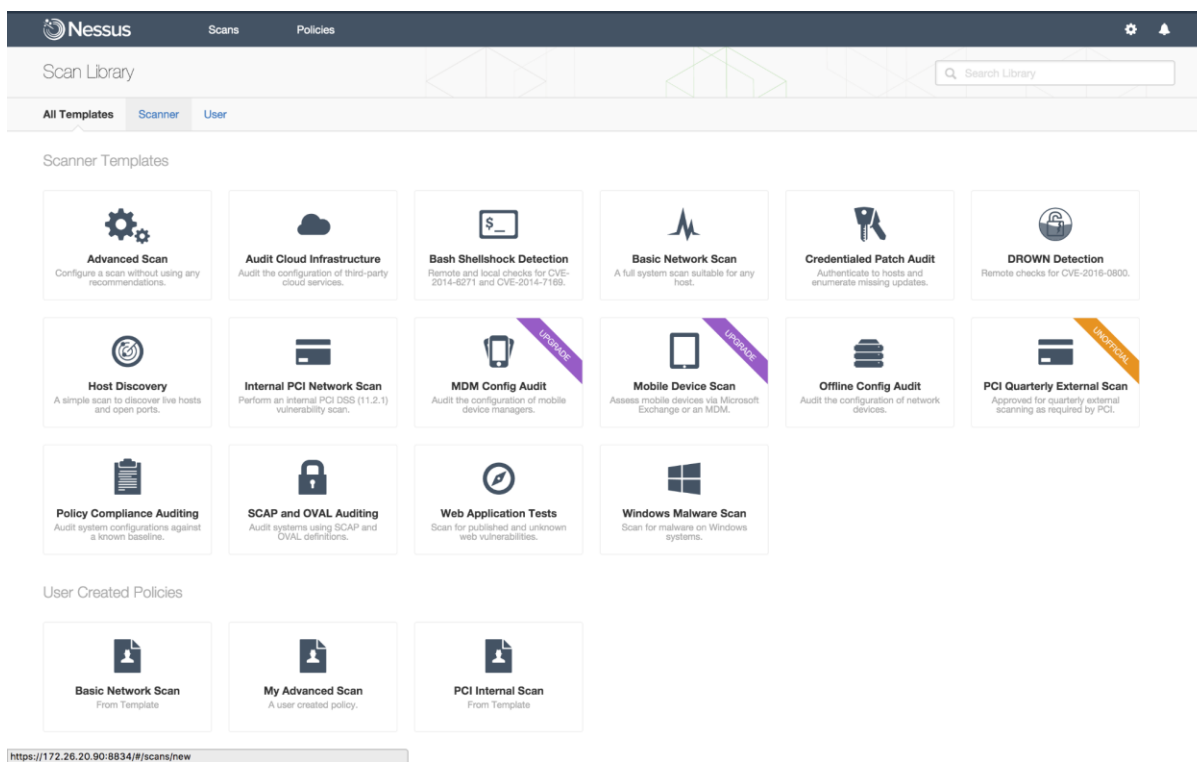


Figure 12 - Nessus Dashboard<sup>12</sup>

Nessus is a tool for remote security scanning. Nessus scans a host/computer and alert us if it discovers any vulnerability that cyber criminals could use to gain access to the computer/system [36]. Nessus runs over 1,200 tests on a computer to determine whether those techniques can be used to a computer and, if so, how much it can harm the system.

Nessus features:

- Easy customization of vulnerabilities or hosts.
- Identifies both remote flaws of hosts as well as missing patches and local flaws.
- Mobile and configuration audits.
- Detects vulnerabilities that a remote attacker can access.

<sup>12</sup> Nessus Scanner Dashboard <[https://external-content.duckduckgo.com/iu/?u=https%3A%2F%2Fwww.tenable.com%2Fsites%2Fdrupal.dmz.tenablesecurity.com%2Ffiles%2Fimages%2Fblog%2FScreen%2520Shot\\_DROWN%2520scan%2520policy%2520template%25202.png&f=1&nofb=1](https://external-content.duckduckgo.com/iu/?u=https%3A%2F%2Fwww.tenable.com%2Fsites%2Fdrupal.dmz.tenablesecurity.com%2Ffiles%2Fimages%2Fblog%2FScreen%2520Shot_DROWN%2520scan%2520policy%2520template%25202.png&f=1&nofb=1)>

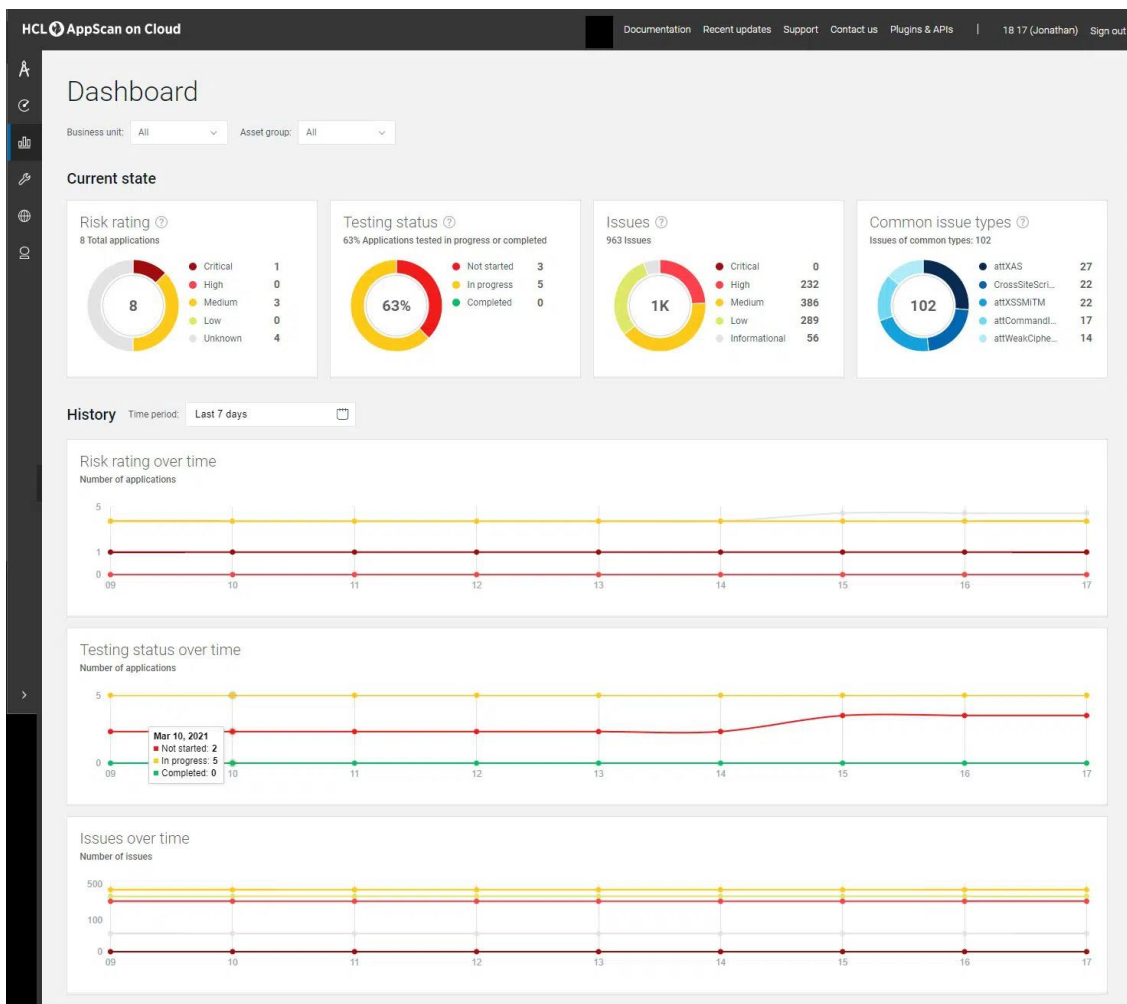


Figure 13 - AppScan Dashboard<sup>13</sup>

AppScan is a tool that can be used by pen-testers and security experts to perform security tests on web applications as well as web services. AppScan is a dynamic analysis testing tool that runs automatic scans that identifies and test applications [37].

AppScan features:

- Automates the target app and tests for vulnerabilities.
- The results are presented in a manner that allows the operator to quickly classify the issues and hone-in on the most critical vulnerabilities found.
- Clear and actionable fix recommendations for each detected issue.

<sup>13</sup> AppScan Dashboard

[https://www.appsecsanta.com/wpcontent/uploads/2022/02/hcl\\_appscan\\_dashboard.jpg](https://www.appsecsanta.com/wpcontent/uploads/2022/02/hcl_appscan_dashboard.jpg)

## Burp-Suite

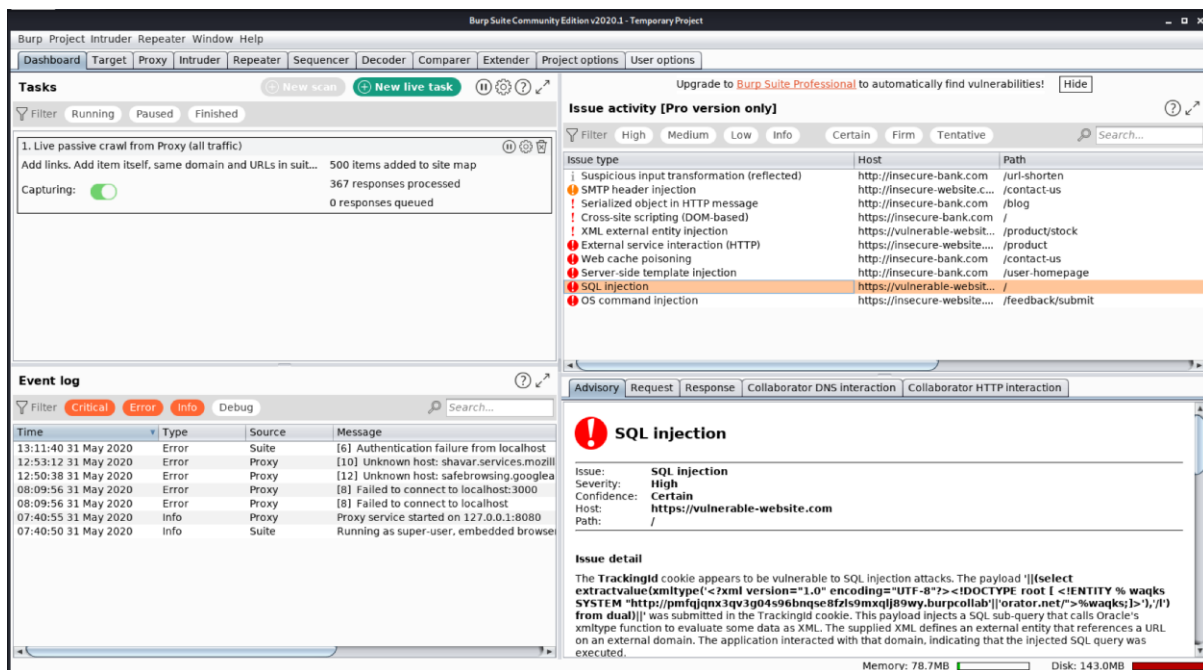


Figure 14 - Burp-Suite Dashboard<sup>14</sup>

Burp-Suite is a graphical tool for performing security testing of web applications. The main usage is to provide solutions for web application security checks [38]. Burp-Suite is split up in three editions:

- Community
- Professional
- Enterprise

Community Edition has the standard functionalities of burp-suite. The other two editions come with a price but provides additional functions to the users. Burp Proxy enables manual testers to intercept all browser-to-target application requests and responses. This tool has a lot of features that a penetration tester can use such as proxy server, scanner, intruder, spider, repeater, decoder, comparer, sequencer, extender API, and clickbandit. Burp-Suite works on Windows, Mac OS and Linux environments.

<sup>14</sup> Burp-Suite Dashboard <<https://www.techpanther.in/2020/06/dashboard-tab-guide-for-burp-suite.html>>



## Kali-Linux

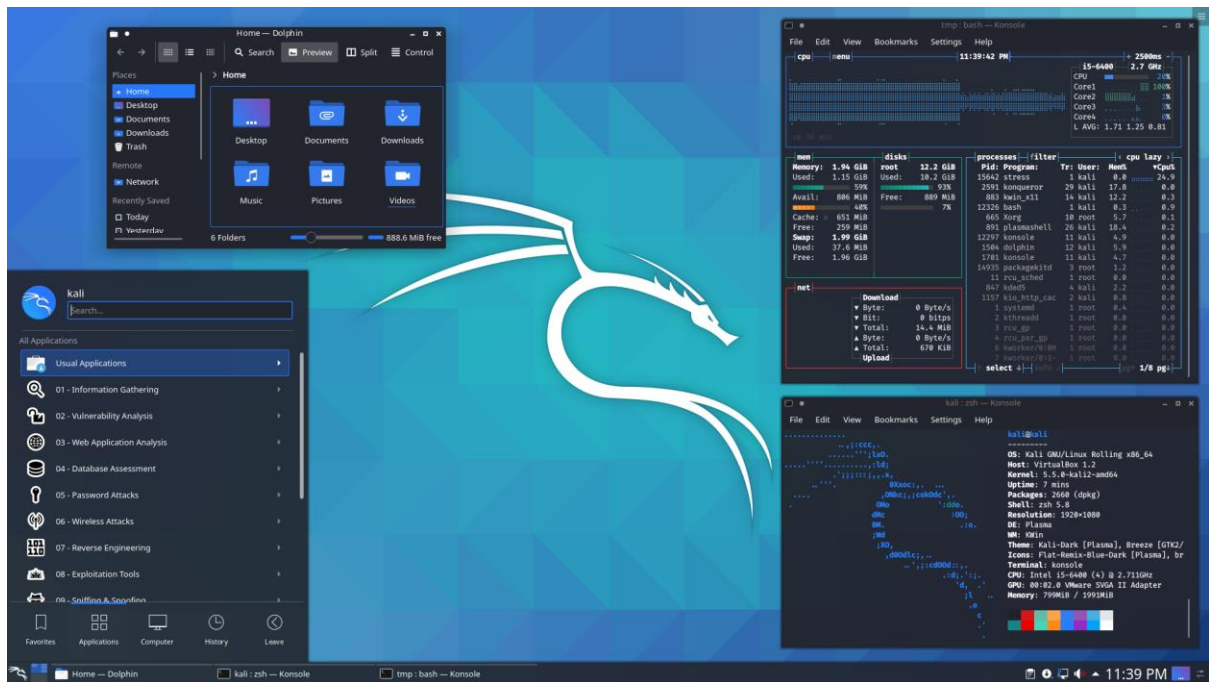


Figure 15 - Kali Linux GUI<sup>15</sup>

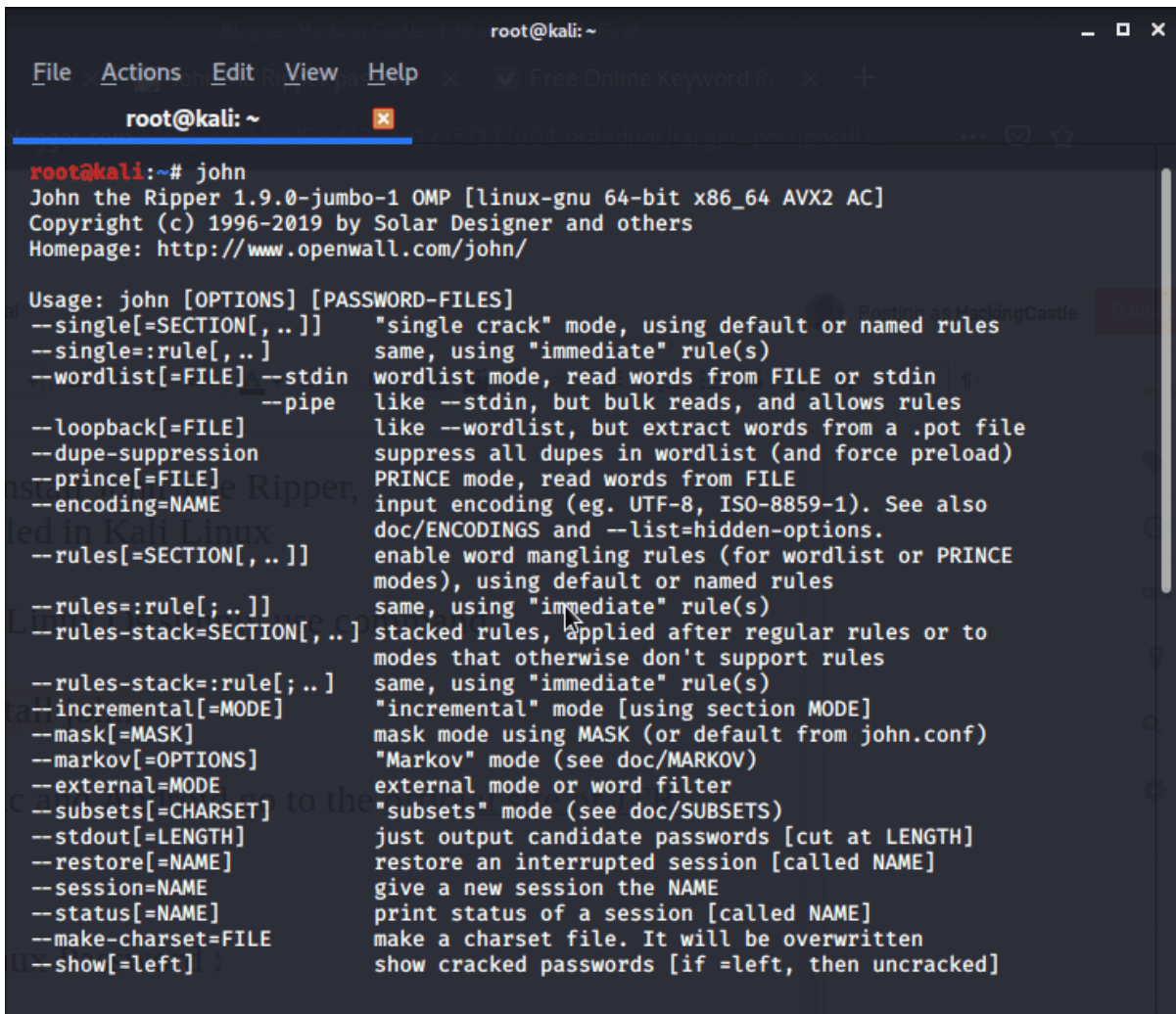
Kali Linux is a tool that is founded by Offensive Security Ltd. It is an open-source pen-testing tool. Kali includes over 600 penetration testing tools that can be used for security tasks.

Kali Linux features:

- Full Disk Encryption (FDE).
- Accessibility features for visually impaired users.
- Full customization of Kali ISOs with live-build allowing us to create our own Kali Linux images.
- It contains Meta package collections that aggregate several toolsets.

<sup>15</sup> Kali-Linux <<https://www.kali.org/blog/kali-linux-2020-2-release/images/release-2020.2-kali-kde-dark.png>>

## John The Ripper

A screenshot of a terminal window on a Kali Linux system. The window title is 'root@kali: ~'. The terminal shows the command 'john' being executed, which displays the version information and copyright for John the Ripper 1.9.0-jumbo-1. Below this, the usage and options for the tool are listed. The options include: --single, --single=:rule, --wordlist, --stdin, --pipe, --loopback, --dupe-suppression, --prince, --encoding, --rules, --rules=:rule, --rules-stack, --rules-stack=:rule, --incremental, --mask, --markov, --external, --subsets, --stdout, --restore, --session, --status, --make-charset, and --show.

```
root@kali: ~
File Actions Edit View Help
root@kali: ~
root@kali:~# john
John the Ripper 1.9.0-jumbo-1 OMP [linux-gnu 64-bit x86_64 AVX2 AC]
Copyright (c) 1996-2019 by Solar Designer and others
Homepage: http://www.openwall.com/john/

Usage: john [OPTIONS] [PASSWORD-FILES]
--single[=SECTION[, ..]] "single crack" mode, using default or named rules
--single=:rule[, ..] same, using "immediate" rule(s)
--wordlist[=FILE] --stdin wordlist mode, read words from FILE or stdin
--pipe like --stdin, but bulk reads, and allows rules
--loopback[=FILE] like --wordlist, but extract words from a .pot file
--dupe-suppression suppress all dupes in wordlist (and force preload)
--prince[=FILE] PRINCE mode, read words from FILE
--encoding=NAME input encoding (eg. UTF-8, ISO-8859-1). See also
doc/ENCODINGS and --list=hidden-options.
--rules[=SECTION[, ..]] enable word mangling rules (for wordlist or PRINCE
modes), using default or named rules
--rules=:rule[; ..]] same, using "immediate" rule(s)
--rules-stack=SECTION[, ..] stacked rules, applied after regular rules or to
modes that otherwise don't support rules
--rules-stack=:rule[; ..] same, using "immediate" rule(s)
--incremental[=MODE] "incremental" mode [using section MODE]
--mask[=MASK] mask mode using MASK (or default from john.conf)
--markov[=OPTIONS] "Markov" mode (see doc/MARKOV)
--external=MODE external mode or word filter
--subsets[=CHARSET] "subsets" mode (see doc/SUBSETS)
--stdout[=LENGTH] just output candidate passwords [cut at LENGTH]
--restore[=NAME] restore an interrupted session [called NAME]
--session=NAME give a new session the NAME
--status[=NAME] print status of a session [called NAME]
--make-charset=FILE make a charset file. It will be overwritten
--show[=left] show cracked passwords [if =left, then uncracked]
```

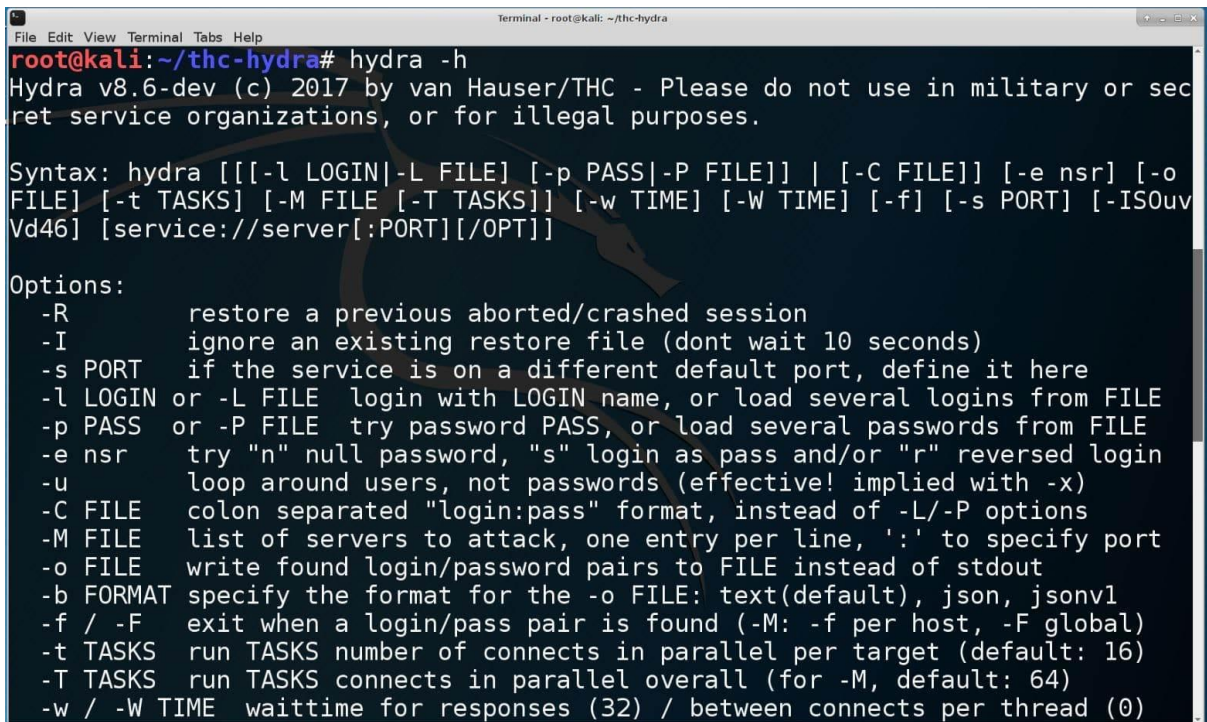
Figure 16 - John The Ripper Options<sup>16</sup>

John The Ripper (JTR) is a free-to-use tool and can be used for password cracking. JTR can even crack very complex passwords. It is one of the most widely used password cracking and testing programs. It is most widely used in dictionary attacks and can detect weak passwords in a network. This tool is available on most platforms such as Unix, Windows, and DOS.

---

<sup>16</sup> John The Ripper (JTR) <<https://1.bp.blogspot.com/-5xyTU5YR234/Xq7K4o7kjRI/AAAAAAAAApY/Ugc5hGShzvsbPDR4LyY6c0gzAOjdaIjAwCLcBGAsYHQ/s1600/john1.png>>

## THC Hydra

A terminal window titled "Terminal - root@kali: ~/thc-hydra" showing the command "hydra -h" and its output. The output includes a warning, a syntax line, and a list of options. The options list includes: -R (restore session), -I (ignore restore file), -s (define port), -l (login name), -p (password), -e (try null password), -u (loop around users), -C (colon separated format), -M (list of servers), -o (write found pairs to file), -b (specify format), -f/-F (exit on success), -t (tasks per target), -T (tasks overall), and -w/-W (waittime).

```
root@kali:~/thc-hydra# hydra -h
Hydra v8.6-dev (c) 2017 by van Hauser/THC - Please do not use in military or sec
ret service organizations, or for illegal purposes.

Syntax: hydra [[[-l LOGIN|-L FILE] [-p PASS|-P FILE]] | [-C FILE]] [-e nsr] [-o
FILE] [-t TASKS] [-M FILE [-T TASKS]] [-w TIME] [-W TIME] [-f] [-s PORT] [-ISOuv
Vd46] [service://server[:PORT][/OPT]]

Options:
-R      restore a previous aborted/crashed session
-I      ignore an existing restore file (dont wait 10 seconds)
-s PORT if the service is on a different default port, define it here
-l LOGIN or -L FILE login with LOGIN name, or load several logins from FILE
-p PASS or -P FILE try password PASS, or load several passwords from FILE
-e nsr  try "n" null password, "s" login as pass and/or "r" reversed login
-u      loop around users, not passwords (effective! implied with -x)
-C FILE colon separated "login:pass" format, instead of -L/-P options
-M FILE list of servers to attack, one entry per line, ':' to specify port
-o FILE write found login/password pairs to FILE instead of stdout
-b FORMAT specify the format for the -o FILE: text(default), json, jsonv1
-f / -F exit when a login/pass pair is found (-M: -f per host, -F global)
-t TASKS run TASKS number of connects in parallel per target (default: 16)
-T TASKS run TASKS connects in parallel overall (for -M, default: 64)
-w / -W TIME waittime for responses (32) / between connects per thread (0)
```

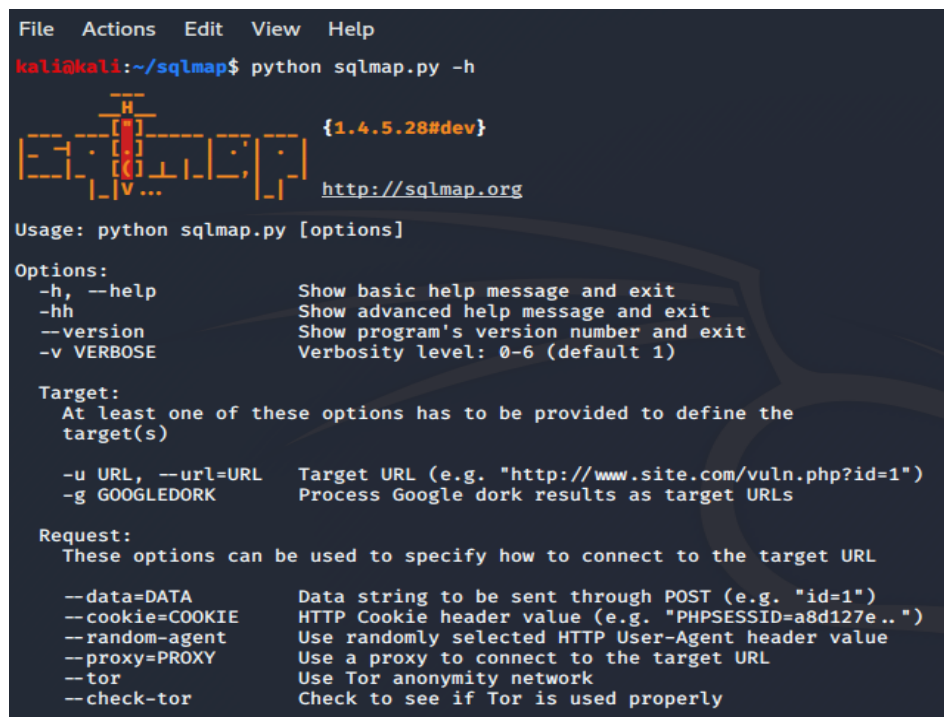
Figure 17 - Hydra Options<sup>17</sup>

Hydra is also a free-to-use tool and can be used for password cracking. It can decrypt passwords from applications and protocols with a dictionary attack [39]. The dictionary attack that applies uses more than 50 protocols including the most popular such as cisco, telnet, FTP, HTTP(S), MySQL, etc. Researchers and security experts can use this tool to detect unwanted access and it works in Kali-Linux, Parrot and other major penetration testing environments.

---

<sup>17</sup> Hydra Options <<https://securityonline.info/wp-content/uploads/2017/05/hydra.jpg>>

## SqlMap



```
File  Actions  Edit  View  Help
kali@kali:~/sqlmap$ python sqlmap.py -h

   H
   |
  [H] {1.4.5.28#dev}
  |
 [.] |
  |
  |
 [.] | [.] | [.] | [.] | [.] | [.] |
  |
  |
 [V] | ...

 http://sqlmap.org

Usage: python sqlmap.py [options]

Options:
  -h, --help                Show basic help message and exit
  -hh                       Show advanced help message and exit
  --version                 Show program's version number and exit
  -v VERBOSE                Verbosity Level: 0-6 (default 1)

Target:
  At least one of these options has to be provided to define the
  target(s)

  -u URL, --url=URL        Target URL (e.g. "http://www.site.com/vuln.php?id=1")
  -g GOOGLEDORK           Process Google dork results as target URLs

Request:
  These options can be used to specify how to connect to the target URL

  --data=DATA              Data string to be sent through POST (e.g. "id=1")
  --cookie=COOKIE          HTTP Cookie header value (e.g. "PHPSESSID=a8d127e..")
  --random-agent            Use randomly selected HTTP User-Agent header value
  --proxy=PROXY            Use a proxy to connect to the target URL
  --tor                     Use Tor anonymity network
  --check-tor              Check to see if Tor is used properly
```

Figure 18 – SQL-Map Options<sup>18</sup>

SQL-Map is a free and open-source penetration testing tool. It can hack over a database server and make that by the automation that it uses to detect and exploit SQL databases [40]. It has a plethora of detection engines and features to make the process easier for the penetration tester.

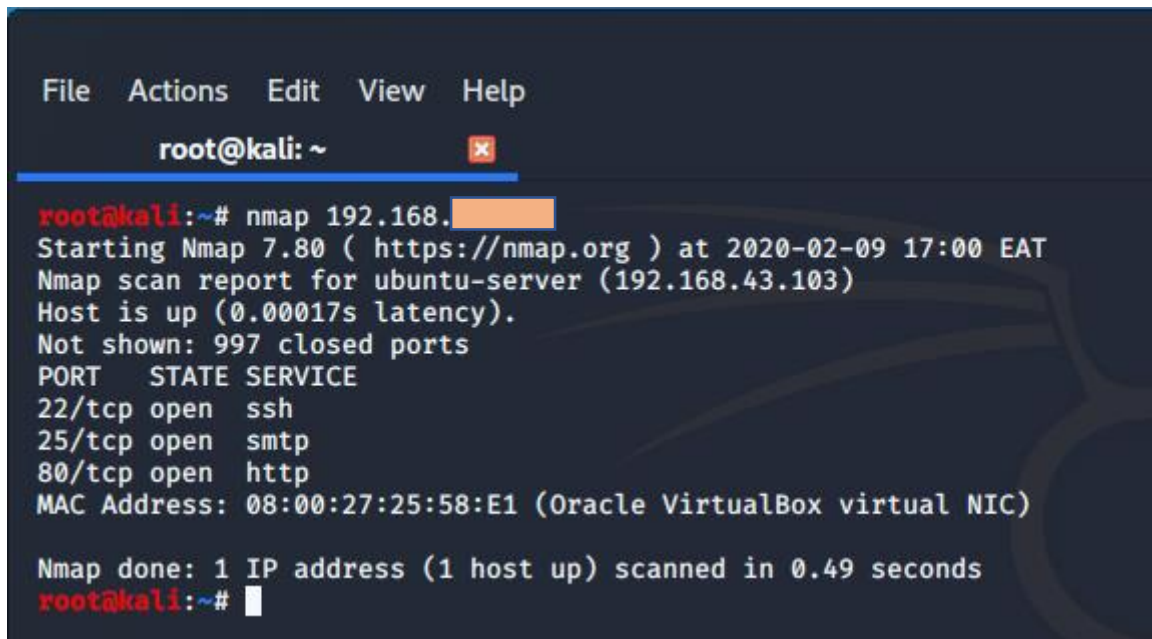
Summarize of SQL-Map features:

- Full support for database management systems such as MySQL, Oracle, PostgreSQL, Microsoft SQL, Microsoft Access, IBM DB2, SQLite, Sybase, SAP MaxDB, HSQLDB, H2, and Informix.
- Full support for SQL injection techniques such as UNION query-based, boolean-based blind, time-based blind etc.
- Supports direct connection to the database without passing via a SQL injection
- Enumeration of users, password hashes, rights, roles, databases, tables, and columns is supported.
- Automatic recognition of password hash formats and support for cracking them using a dictionary-based attack
- Supports dump database tables entirely or specific columns as per user's choice
- Supports search for specific database names, tables, or columns across all databases' tables
- Support TCP connections between the attacker machine and the database server.

---

<sup>18</sup> SQL- Map Options <<https://sectechno.com/wp-content/uploads/2020/05/sqlmap-1.png>>

## NMap



```
File Actions Edit View Help
root@kali: ~
root@kali:~# nmap 192.168.43.103
Starting Nmap 7.80 ( https://nmap.org ) at 2020-02-09 17:00 EAT
Nmap scan report for ubuntu-server (192.168.43.103)
Host is up (0.00017s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
25/tcp    open  smtp
80/tcp    open  http
MAC Address: 08:00:27:25:58:E1 (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 0.49 seconds
root@kali:~#
```

Figure 19 - Nmap Scan<sup>19</sup>

Nmap is a free and open-source scanning tool for networks. It works on many platforms such as Windows, Linux, Mac OS, etc. Nmap determines:

- Hosts that are available on the network
- Services that the hosts are offering
- Operating Systems
- Versions that run
- Packets of filters and firewalls that are in use
- Open ports

Features of Nmap:

- Discovers hosts on a network
- Identifies open ports on target hosts
- It is used to determine network inventory, network mapping, maintenance, and asset management
- To find and exploit vulnerabilities in a network
- It generates traffic to hosts on a network, response analysis and response time measurement

---

<sup>19</sup> Nmap <<https://linuxide.com/wp-content/uploads/2020/02/1.-nmap-basic-usage.png>>

## Wireshark

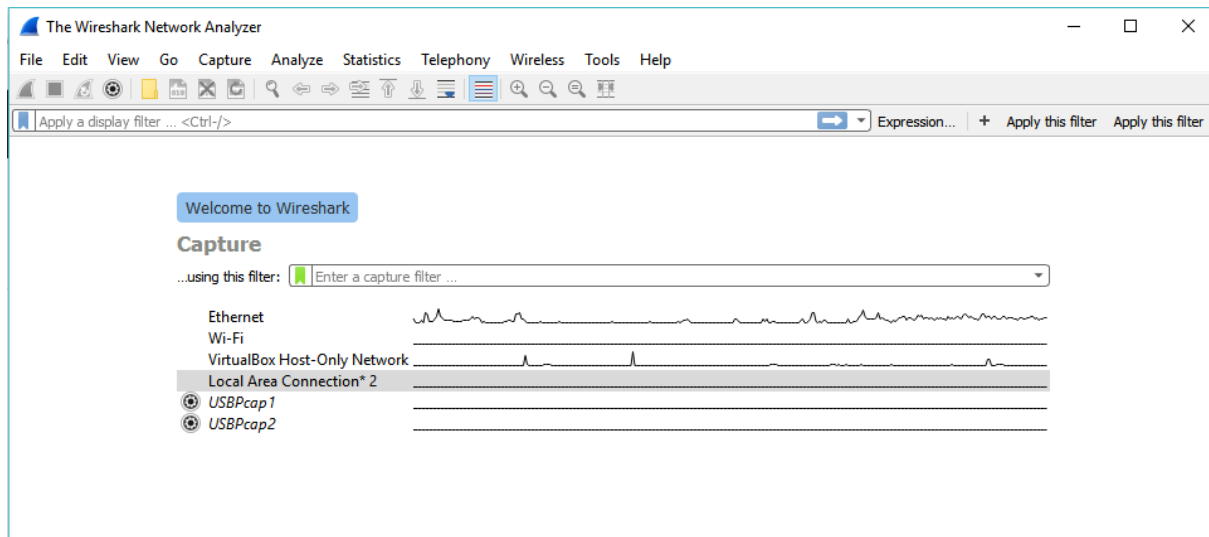


Figure 20 - Wireshark Dashboard<sup>20</sup>

Wireshark is one of the well-known tools. It is free to use penetration testing tool. More specifically, it is a network protocol analyzer, it captures the traffic that flows on a computer network. Wireshark can be installed in Windows, Unix, Mac OS, and many others.

Wireshark features [41]:

- Comprehensive analysis of hundreds of protocols
- Live capture and offline analysis
- Standard three-pane packet browser
- Multi-platform: Runs on Windows, Linux, OS X, FreeBSD, NetBSD, and many others
- Captured network data can be viewed using a graphical user interface (GUI) or the TTY-mode TShark tool
- Rich VoIP analysis
- Read/write file-formats
- Capture files that have been compressed using gzip can be decompressed on the fly. Many protocols, including IPsec, ISAKMP, Kerberos, SNMPv3, SSL/TLS, WEP, and WPA/WPA2, are supported for decryption.

---

<sup>20</sup> Wireshark Dashboard <<https://sw4pn1lp.github.io/wireshark/images/1gui.png>>

## Netsparker

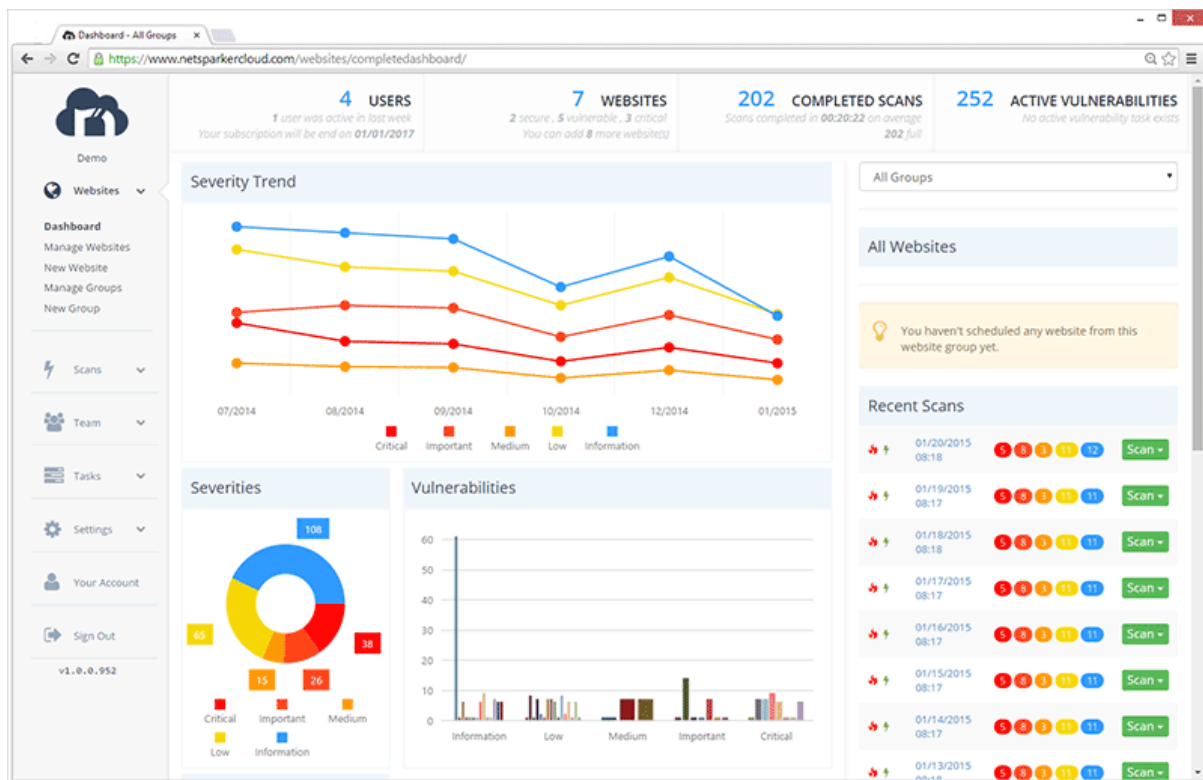


Figure 21 - Netsparker Dashboard<sup>21</sup>

Netsparker is a web application security scanner. It is used to identify SQLi and Cross-Site Scripting (XSS) vulnerabilities in websites, services, and applications. This tool is precise with the results because it produces a Proof of Concept to confirm that the results are not false positives [42].

Netsparker features:

- Reporting and Exploitation.
- Comprehensive Scanning.
- Flexible Deployment Options.
- Advanced Website Crawling Technologies.
- Easy to Configure Authentication.

<sup>21</sup> Netsparker Dashboard <<https://www.nss.gr/wp-content/uploads/2018/06/netsparker-cloud-dashboard.png>>

## Zed Attack Proxy

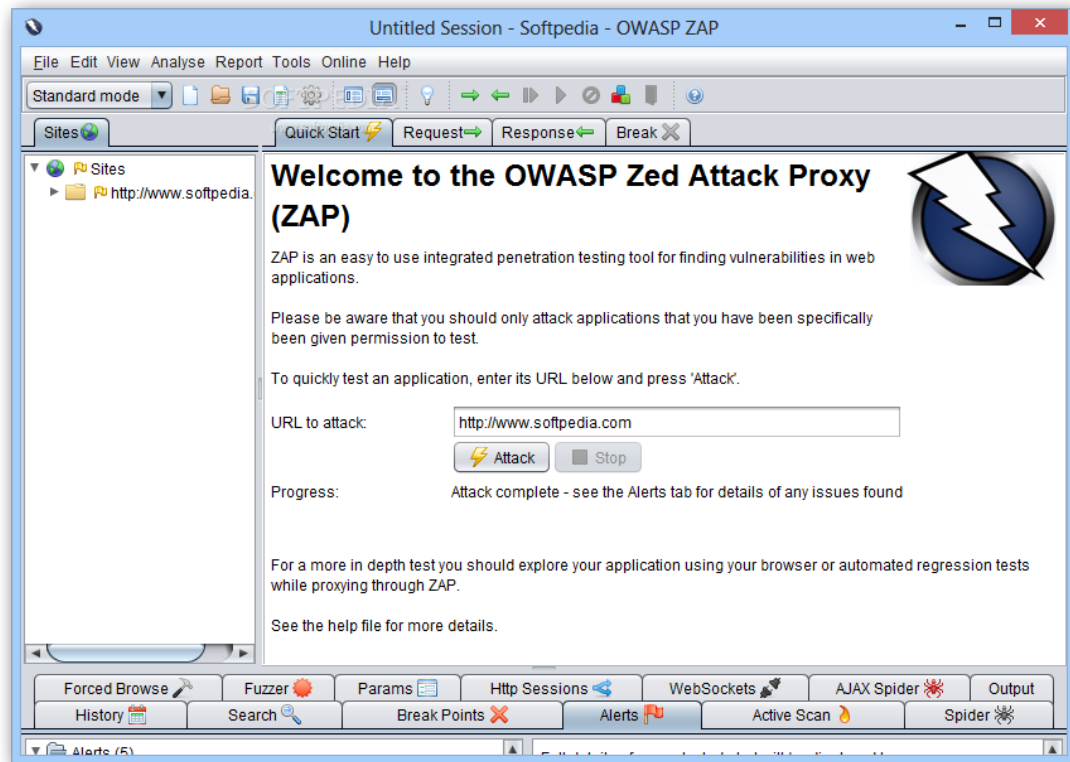


Figure 22 - Zed Attack Proxy Dashboard<sup>22</sup>

Zed attack proxy is a tool that finds security vulnerabilities in web applications. ZAP is a freely available tool and provides automated scanners and tools that find security vulnerabilities manually. It can be used by either professional penetration testers or someone that is new in the field of application security.

Features of ZAP:

- Intercepting Proxy
- Automated Scanner
- Brute Force Scanner
- Fuzzing
- Port Scanning
- WebSockets
- Advanced SQL Injection Scanner
- Advanced Alerts

---

<sup>22</sup> Zed Attack Proxy Dashbaord <[http://www.testingtoolsguide.net/wp-content/uploads/2016/11/OWASP-ZAP\\_1.png](http://www.testingtoolsguide.net/wp-content/uploads/2016/11/OWASP-ZAP_1.png)>



## Nexpose



Figure 23 - Nexpose Dashboard<sup>23</sup>

Nexpose is another tool for vulnerability scanning. The user can use this tool via a browser. This tool is used to scan the vulnerability of a network. Nexpose can scan open ports, services, and running applications. It supports vulnerability management's lifecycle, including verification, impacts analysis, discovery, risk classification, detection, reporting, and mitigation.

Features of Nexpose [43]:

- Real Risk Score
- Adaptive Security
- Policy Assessment
- Remediation Reporting
- Integration with Metasploit

<sup>23</sup> Nexpose Dashboard <<https://samehsabry.files.wordpress.com/2016/07/nexpose-dashboard.png>>

## Core Impact

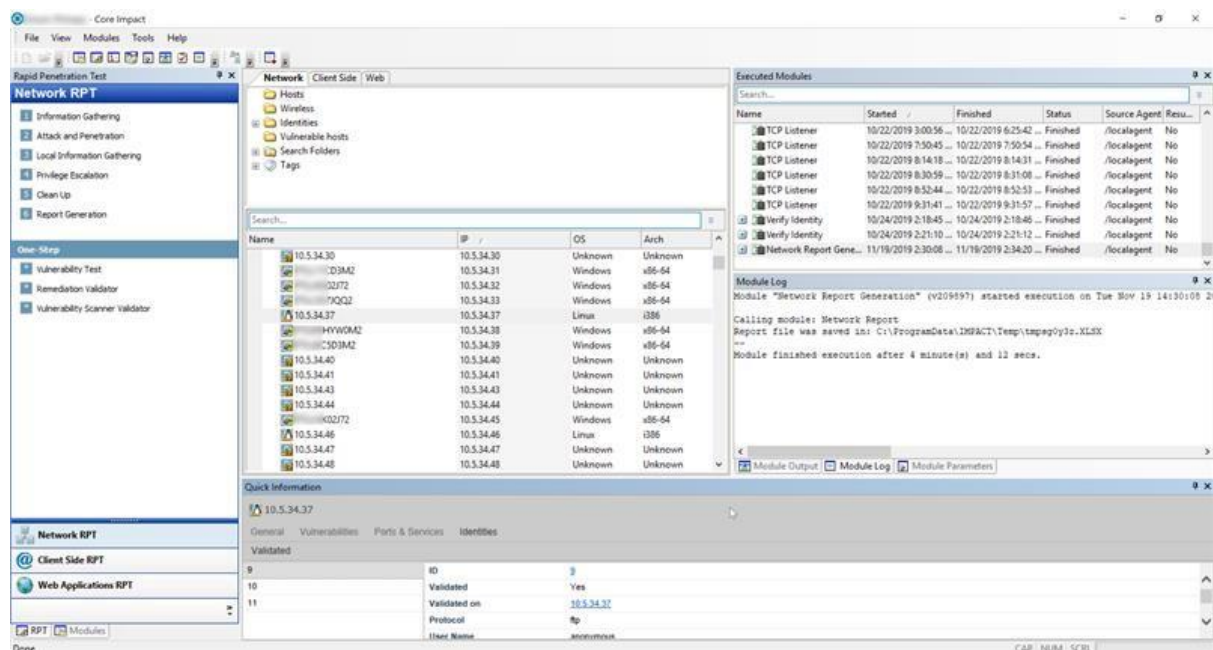


Figure 24 - Core Impact Dashboard<sup>24</sup>

Core Impact is a penetration testing platform developed to make it simple for security teams to run complex testing. You can securely test your environment using the same approaches as of today's attackers with guided automation and approved exploits.

Features of Core Impact [44]:

- Guided Automation. Core Impact's Rapid Penetration Tests (RPTs) are accessible automation designed to automate common and repetitive tasks.
- Certified Exploits
- Patented Agents
- Robust Error Prevention
- Teaming
- Reporting

<sup>24</sup> Core Impact Dashboard <<https://gdm-catalog-fmapi-prod.imgix.net/ProductScreenshot/a5e5196e-d92c-4d60-9a8d-ee0e9d49b077.jpeg>>

The table below lists some penetration testing tools that can be used in penetration testing, as well as information such as the phase to be used and the operating system it can run.

No	Name	Type	Phase that can be used	License	Operating System
1	Metasploit	Vulnerability scanner and exploit	Phase 3 & 4	No	Cross-platform
2	w3af	Web vulnerability scanner	Phase 2 & 2	No	Cross-platform
3	Nessus	Vulnerability scanner	Phase 2 & 3	Both	Cross-platform
4	AppScan	Web vulnerability scanner	Phase 2 & 3	Yes	Windows
5	Burp-Suite	Web vulnerability scanner	Phase 2 & 3 & 4	Both	Cross-platform
6	Kali Linux	Collection of various tools	Phase 2 & 3	No	Linux
7	John The Ripper	Password cracking	Phase 4	No	Cross-platform
8	THC Hydra	Password cracking	Phase 4	No	Cross-platform
9	SqlMap	Detecting and exploiting SQL injection	Phase 4	No	Linux, Windows, and Mac OS
10	NMap	Network Mapper	Phase 2 & 3	Both	Cross-platform
11	Wireshark	Network protocol analyzer	Phase 2 & 3	Both	Cross-platform
12	Netsparker	Web vulnerability scanner	Phase 2 & 3	Both	Cross-platform
13	Zed Attack Proxy	Web vulnerability scanner	Phase 2 & 3	No	Cross-platform
14	Nexpose	Entire vulnerability management lifecycle	Phase 2 & 3 & 5	Both	Linux, Windows
15	Core Impact	Vulnerability scanner and exploit	Phase 2 & 3 & 4 & 5	Yes	Windows

Table 1 – Tools for Penetration Testing

## 4. Critical Infrastructures

As we mentioned before, it is a priority to protect the critical infrastructures. Thus, here is the Critical Infrastructure that we tested, but before we start getting information and analyze the networks we need to understand what is the SDN Controller and Charging Stations for Electrical Vehicles. Below we will give a brief example of how they work and some advantages and disadvantages that may have.

### 4.1 *Software-Defined Network/SDN controller*

SDN stands for Software-Defined Networking and the SDN controller is a software that controls the network traffic in order to enhance management of the network and application performance. SDN controller platforms are frequently placed on servers, so the controller utilizes protocols to guide switches on where to send packets. [45] [46].

The SDN controller sends traffic based on forwarding policies defined by a network operator; as a consequence, manual settings for particular network devices are reduced.

The centralized controller supports automated network administration and makes it simpler to connect and administrate business applications by removing the control plane from network hardware and executing it as software.

In fact, the SDN controller acts as a network's operating system (OS).

The controller's role is critical in a software-defined network. It is located between network devices on one end and network applications on the other. Any communication between apps and network devices must go through the controller.

The controller communicates with some other applications in the network such as firewalls and load balancers. The controller also communicates with individual network devices via a southbound interface, which is often used via the OpenFlow protocol. The controller can use southbound protocols to configure network devices and select the optimum network path for application traffic.

#### Advantages of SDN controller

- Security: The controller that SDN has can provide security to the entire network. That means this controller guarantees that it will follow some regulations and information in the network. SDN also provides a centralized administration system. A single entity will be in charge of security and features. This system is considered to be exceptionally secure since it only requires one central point. The administrator on the other hand, might simply block security threats from accessing the system.
- Centralization: SDN enables centralized network administration. All the devices on the network can be either monitored or managed from a single place.
- Scalability: The SDN provides scalability which means that the network can be modified at any time without the need for any purchase or configuration resources.

- Optimization: SDN is capable of allocating roles in the network, as a result there is no limitation on devices engaged in a task.

#### Disadvantages of SDN controller

- Latency: Each device connected to a network takes up some space on it, so the more virtualized devices from the SDN there are on a network, the more latency a network will encounter.
- Maintenance: Maintenance is a critical part of networking in order to carry out its activities. Because the SDN is lacking in such cases, managing such devices makes it very hard. Especially when expanding up a network.
- Complexity: SDN security mechanisms are not well defined. Despite the presence of certain third-party service providers, security vulnerabilities exist. Only individuals with expertise in managing SDN networks can prevent major attacks.
- Configuration: The process of reconfiguring an SDN network is time-consuming and expensive. SDN protocols and controllers, in particular, cannot be implemented by configuring each one separately.

## ***4.2 Charging Infrastructures for Electrical Vehicles***

In a mission to find sustainable energy to rely on, EVs came to save the day by using only batteries and sustainable energy. The three main components of an electric vehicle are [47]:

1. the controller
2. the battery
3. and the electric motor

As we mention above, these cars work with batteries, which means electricity is provided by them. Electric vehicles emit zero carbon dioxide. However, if the electricity used to charge the batteries is supplied by the normal power system, carbon dioxide emissions are not considerably decreased.

The way that the car can charge is separated into two parts:

- parking charges and
- ongoing charges

The difference between those two is that the Parking charge is all the places that you can park your EV for some time, such places are houses, offices, parking lots or centres, etc.

On the other hand, an ongoing charge is a quick method for charging EVs that seeks to facilitate charging on long-distance excursions and serves a similar purpose as petrol stations. Because of the increasing urge to reduce emissions European Union trying to bring more EVs to the roads. EU traffic is evolving, with considerable improvements being implemented to fulfil the need for ecologically friendly travel.

Electric vehicles are gaining popularity because they emit no pollutants, are silent, and are three times more efficient than gasoline engines.

As the usage of electric vehicles grows, the Electricity Distribution Networks will encounter local issues such as overloading the local network and affecting the mains voltage. An optimistic solution is "smart charging". With smart charging, we avoid overloading the network by supporting the network when needed.

Some of the benefits that smart charging offers are:

1. Price: because most people charge the EV vehicle either at home or in their offices the price that they pay for electrical installation is high. Users will be able to charge their cars at home using smart charging without having to consider the demands of their electrical system. Users will be able to take advantage of inexpensive costs in the morning while demand is low with this strategy.
2. Flexibility: Smart charging enables EVs to be flexible loads dispersed over the network, which the Network Operator may use to fulfil the demands of the network and so avoid costly network expansions.
3. Sustainable: Because of the low cost of charging obtained by smart charging, as well as the efficiency of electric cars, the usage of primary energy will be substantially reduced, resulting in a significant decrease in emissions. The flexibility provided by smart charging facilitates the deployment of dispersed RES systems and enables higher penetration with many benefits for all users.

Despite, the fact that smart charging allows having a big advantage to drive an electric car, it has potential threats and attacks that must be pointed out. Many attacks, for example, take advantage of flaws in-network services such as SSH, HTTP, SOAP, and others that have an inadequate infrastructure.

Apart from that, these services are also accessible via mobile phones. As a result, these services can be compromised even without physical access through the mobile network interface. Attackers will attempt to overcome these systems by using services with no or very outdated encryptions and gaining access to the system to install malicious code that will be used to damage the electrical grid infrastructure or, more commonly, to gain access to the payment system in order to extract money from it.

Another possible attack is an SQL injection attack. In this scenario, the attacker will gain access to the database of the company. In this way, he would have complete access to all the enterprise's recorded data, with the purpose of corrupting it. Doing so will provoke it to become useless and irreparably damaged [48] & [49].

## 5. Results of the analysis of Critical Infrastructures

In this section of the thesis, we analyze the security in Critical Infrastructures by applying various tools mentioned above. In this case, we use tools like Nmap and Nessus to analyze the open ports and their severity to discover various vulnerabilities in the network we have. Continuing with the mapping process, we will proceed by exploiting any vulnerabilities using tools such as Metasploit. The domains in which the analysis was done were,

- SDN Controller and
- Charging Infrastructure for Electrical Vehicles

The results of the analysis are shown and discussed below.

### 5.1 Analyzing the SDN Controller

Initially, to start the process properly we will reconnaissance the network to identify any open ports, thereupon we will do a vulnerability identification to identify all the vulnerabilities of the network and lastly, we will exploit the vulnerabilities if possible. In this chapter we will analyse the SDN network and we will separate it in to 3 stages as explained:

1. Reconnaissance
2. Vulnerability identification
3. Exploitation

#### 5.1.1 Reconnaissance Phase of SDN Controller

Starting the process, we search for any and every information that we can obtain. In this phase, we use a mapper tool named Nmap to scan the network. The command to run Nmap and make a suitable scan is shown below.

```
# Nmap 7.92 scan initiated Wed Mar  2 06:44:36 2022 as: nmap -sS -T4 -p- -oN nmap-
full-portscan.txt [REDACTED]
Nmap scan report for [REDACTED] ([REDACTED])
Host is up (0.085s latency).
Not shown: 65522 closed tcp ports (reset)
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
443/tcp   open  https
2181/tcp  open  eforward
2375/tcp  open  docker
3181/tcp  open  bmcpatrolagent
8889/tcp  open  ddi-tcp-2
10443/tcp open  cirrossp
16633/tcp open  unknown
20443/tcp open  unknown
26633/tcp open  unknown
30443/tcp open  unknown
36633/tcp open  unknown

# Nmap done at Wed Mar  2 06:46:32 2022 -- 1 IP address (1 host up) scanned in
115.86 seconds
```

Figure 25 - Results of the Nmap in SDN Controller

- sS: TCP SYN port scan.
- T4: Aggressive (4) speeds scans. We use that when we have a reasonably fast and reliable network.
- -p-: Port scan all ports.
- -oN: Normal output to the file full-portscan.txt

As the above results show, we have a lot of options to begin with. After analyzing the results and searching for every port, we continue to look for a vulnerability to get access to. The docker is a port that we might be able to exploit and gain root access.

### 5.1.2 Vulnerability Identification of SDN Controller

Consequently, we are searching for vulnerabilities in the network with the Nessus tool to find the severity of these ports. The results of the Nessus are shown below.

Table of Nessus:

Critical	High	Medium	Low	Info
2	0	5	0	39

SEVERITY	CVSS V3.0	PLUGIN	NAME
Critical	9.4	150154	nginx 0.6.x < 1.20.1 1-Byte Memory Overwrite RCE
Critical	10.0	124029	Docker Remote API Detection
Medium	6.5	51192	SSL Certificate Cannot Be Trusted
Medium	6.5	104743	TLS Version 1.0 Protocol Detection
Medium	6.1	136929	JQuery 1.2 < 3.5.0 Multiple XSS
Medium	5.3	15901	SSL Certificate Expiry
Medium	6.4*	57582	SSL Self-Signed Certificate
Info	N/A	10114	ICMP Timestamp Request Remote Date Disclosure
Info	N/A	45590	Common Platform Enumeration (CPE)



Info	N/A	54615	Device Type
Info	N/A	84502	HSTS Missing <u>From</u> HTTPS Server
Info	N/A	10107	HTTP Server Type and Version
Info	N/A	12053	Host Fully Qualified Domain Name (FQDN) Resolution
Info	N/A	24260	<del>HyperText</del> Transfer Protocol (HTTP) Information
Info	N/A	106658	JQuery Detection
Info	N/A	11219	Nessus SYN scanner
Info	N/A	19506	Nessus Scan Information
Info	N/A	11936	OS Identification
Info	N/A	117886	OS Security Patch Assessment Not Available
Info	N/A	50845	OpenSSL Detection
Info	N/A	66334	Patch Report
Info	N/A	70657	SSH Algorithms and Languages Supported
Info	N/A	149334	SSH Password Authentication Accepted
Info	N/A	10881	SSH Protocol Versions Supported
Info	N/A	153588	SSH SHA-1 HMAC Algorithms Enabled
Info	N/A	10267	SSH Server Type and Version Information
Info	N/A	56984	SSL / TLS Versions Supported
Info	N/A	45410	SSL Certificate 'commonName' Mismatch
Info	N/A	10863	SSL Certificate Information
Info	N/A	70544	SSL Cipher Block Chaining Cipher Suites Supported

Info	N/A	21643	SSL Cipher Suites Supported
Info	N/A	57041	SSL Perfect Forward Secrecy Cipher Suites Supported
Info	N/A	94761	SSL Root Certification Authority Certificate Information
Info	N/A	156899	SSL/TLS Recommended Cipher Suites
Info	N/A	22964	Service Detection
Info	N/A	25220	TCP/IP Timestamps Supported
Info	N/A	84821	TLS ALPN Supported Protocol Enumeration
Info	N/A	87242	TLS NPN Supported Protocol Enumeration
Info	N/A	62564	TLS Next Protocols Supported
Info	N/A	121010	TLS Version 1.1 Protocol Detection
Info	N/A	136318	TLS Version 1.2 Protocol Detection
Info	N/A	110723	Target Credential Status by Authentication Protocol - No Credentials Provided
Info	N/A	10287	Traceroute Information
Info	N/A	11154	Unknown Service Detection: Banner Retrieval
Info	N/A	10386	Web Server No 404 Error Code Check
Info	N/A	106375	nginx HTTP Server Detection

*Table 2 - Nessus Results of SDN Controller*

\* Indicates the v3.0 score was not available; the v2.0 score is shown

Above are the Nessus results for the SDN controller, notice that there are 2 results where they are Critical, 5 are Medium Severity and the rest are Info.

Starting from the Critical severity, we found:

- Docker and
- nginx 0.6.x

The docker vulnerability has an API exposed. As a result, a remote attacker who is not authorized might execute the administrator Docker command. Based on Nessus, we can configure API access controls to overcome this problem.

The nginx 0.6.x version vulnerability affected by a remote code execution vulnerability. In this case, an unauthenticated remote attacker could cause 1-byte memory to overwrite by using a specially crafted DNS response, which will result in worker process crash or, potentially, in arbitrary code execution. A way to solve this problem is to upgrade the nginx to 1.20 or later edition.

Continuing with the medium risks of SDN Controller, there are 5 different problems that we are going to point out.

- SSL Certificate Cannot Be Trusted
- SSL Self-Signed Certificate
- TLS Version 1.0 Protocol Detection
- SSL Certificate Expiry
- JQuery 1.2 < 3.5.0 Multiple XSS

#### SSL Certificate Cannot Be Trusted

The “SSL certificate cannot be trusted” can occur in three different ways and that might break the chain of trust. Below we can see the three cases.

- First, the server's certificates may not come from a known public certificate authority. This can occur when the public certificate authority is an unrecognized or self-signed certificate.
- In the second case, a certificate that is no longer valid at the time of the scan may be included in the certificate chain. This can happen if the scan runs before or after one of the certificate's 'notBefore' or 'notAfter' dates.
- In the third case, the signature that a certificate has may not match the information on the certificate. Such signatures can be repaired by re-signing the certificate.

A solution to this problem is to purchase or to create a proper SSL certificate for the service.

#### TLS Version 1.0 Protocol Detection

TLS version 1.0 allows remote connection. However, this version has several weaknesses in its encryption. Newer versions such as 1.2 or 1.3 solve this problem and should be used whenever possible.

The solution to that is to enable support for TLS 1.2 and 1.3 and disable support for TLS 1.0.

### JQuery 1.2 < 3.5.0 Multiple XSS

The remote web server's JQuery version is more than or equal to 1.2 and less than 3.5.0. As a result, it is vulnerable to a variety of cross-site scripting flaws.

It should be noted that the vulnerabilities listed in this plugin have no security consequences for PAN-OS.

Now, the solution to this is to upgrade the JQuery to version 3.5.0 or later.

### SSL Certificate Expiry

This is simple to the synopsis; the remote server's SSL certificate has already expired. That means that the plugin has checked the expiry dates of the certificates and made some reports on whether any have already expired. The solution to that is to purchase or generate a new SSL certificate to replace the existing one.

### SSL Self-Signed Certificate

The problem with the SSL certificate is that it results in an unrecognized self-signed certificate. The X.509 certificates are not signed by the certificate authority.

It is important to note that this plugin does not look for certificate chains that terminate in a certificate that is not self-issued but is signed by an unknown certificate authority.

A solution to that is to purchase or generate a proper SSL certificate for this service.

### **5.1.3 *Exploitation Phase of SDN Controller***

While analyzing the docker port in SDN we were able to identify that the docker port (port 2375) was open and can be exploited because of the exposed API. The way that we compromise this port was quite easy.

The first step was to identify if the API was able to receive my commands. In order to enumerate the docker API we use the curl command as shown below.

```

kali@kali:~/Documents/Thesis$ curl -s http://[REDACTED]:2375/version | jq
{
  "Platform": {
    "Name": ""
  },
  "Components": [
    {
      "Name": "Engine",
      "Version": "20.10.7",
      "Details": {
        "ApiVersion": "1.41",
        "Arch": "amd64",
        "BuildTime": "2021-10-22T00:45:53.000000000+00:00",
        "Experimental": "false",
        "GitCommit": "20.10.7-0ubuntu5~20.04.2",
        "GoVersion": "go1.13.8",
        "KernelVersion": "5.4.0-107-generic",
        "MinAPIVersion": "1.12",
        "Os": "linux"
      }
    },
    {
      "Name": "containerd",
      "Version": "1.5.9-0ubuntu1~20.04.4",
      "Details": {
        "GitCommit": ""
      }
    },
    {
      "Name": "runc",
      "Version": "1.0.1-0ubuntu2~20.04.1",
      "Details": {
        "GitCommit": ""
      }
    },
    {
      "Name": "docker-init",
      "Version": "0.19.0",
      "Details": {
        "GitCommit": ""
      }
    }
  ],
  "Version": "20.10.7",
  "ApiVersion": "1.41",
  "MinAPIVersion": "1.12",
}

```

Figure 26 - Information about the Docker

```

"Version": "20.10.7",
"ApiVersion": "1.41",
"MinAPIVersion": "1.12",
"GitCommit": "20.10.7-0ubuntu5~20.04.2",
"GoVersion": "go1.13.8",
"Os": "linux",
"Arch": "amd64",
"KernelVersion": "5.4.0-107-generic",
"BuildTime": "2021-10-22T00:45:53.000000000+00:00"
}

```

Figure 27 - Information about the Docker

With the command “curl -s http://open.docker.socket:2375/version | jq” we get a response from the API with the version and some more information about the API.

As long as the API response to our command we are able to compromise the API with docker commands.

```
└─# docker -H [REDACTED]:2375 run --rm -it --privileged --net=host -v /:/mnt alpine
cat /mnt/etc/shadow
/# whoami
root
/# ls
bin      etc      lib      mnt      proc     run      srv      tmp      var
dev      home    media    opt      root     sbin     sys      usr
```

Figure 28 - Privilege Escalation on the Docker

Because we were able to get root in the SDN network we were able also to modify or change anything in the docker.

## 5.2 Analyzing the Charging Infrastructure for Electrical Vehicles

In this chapter, we continue with the Charging Infrastructure of Electrical Vehicles. As we did in the previous chapter we will follow the same steps here.

1. Reconnaissance
2. Vulnerability Identification
3. Exploitation

By doing that we will be able to understand the infrastructure in depth.

### 5.2.1 Reconnaissance Phase of Charging Infrastructure for Electrical Vehicles

With the nmap scan we map the network to find any open ports that we might be able to compromise. In this infrastructure, we didn't have a lot of information to work with. The only information from the mapping process is that we have an SSH as an open port.

```

Nmap scan report for [redacted]
Host is up (0.079s latency).
Not shown: 65534 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.6p1 Ubuntu 4ubuntu0.6 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 b9:78:e5:cf:85:76:d7:bc:fa:5f:52:6d:ed:ef:f6:53 (RSA)
|   256  ff:40:34:78:2e:bd:07:0e:c6:16:0b:c0:69:b0:7d:05 (ECDSA)
|_  256  43:46:dc:7d:9c:63:86:2f:c4:2c:8e:63:7a:9e:a3:5c (ED25519)
No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/submit/ ).
TCP/IP fingerprint:
OS:SCAN(V=7.92%E=4%D=6/27%OT=22%CT=1%CU=35094%PV=Y%DS=1%DC=I%G=Y%TM=62B98D5
OS:2%P=x86_64-pc-linux-gnu)SEQ(SP=107%GCD=1%ISR=10D%TI=Z%TS=A)OPS(O1=M556ST
OS:11NW7%O2=M556ST11NW7%O3=M556NNT11NW7%O4=M564ST11NW7%O5=M556ST11NW7%O6=M5
OS:56ST11)WIN(W1=FE88%W2=FE88%W3=FE88%W4=FE88%W5=FE88%W6=FE88)ECN(R=Y%DF=Y%
OS:T=40%W=FAF0%O=M556NNSNW7%CC=Y%Q=)T1(R=Y%DF=Y%T=40%S=0%A=S+%F=AS%RD=0%Q=)
OS:T2(R=N)T3(R=N)T4(R=N)T5(R=Y%DF=Y%T=40%W=0%S=0%A=S+%F=AR%O=%RD=0%Q=)T6(R=
OS:N)T7(R=N)U1(R=Y%DF=N%T=40%IPL=164%UN=0%RIPL=G%RID=G%RIPCK=G%RUCK=G%RUD=G
OS:))IE(R=N)

Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE
HOP RTT ADDRESS
1 79.34 ms [redacted]

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 1011.40 seconds

```

Figure 29 - Nmap results of the Charging Infrastructure

The Nmap commands was the same as the previous. "nmap -sS -T4 -p-".

Because we were not able to get enough information, we continued to the vulnerability scanner to scan the network in depth.

### 5.2.2 Vulnerability Identification Phase of the Charging Infrastructure

Consequently, with the same process as before we make a full scan with Nessus and we find that there is a Medium Severity. The results of the scans are shown below.

Table of Nessus:

Critical	High	Medium	Low	Info
0	0	1	0	16

SEVERITY	CVSS V3.0	PLUGIN	NAME
Medium	5.0*	12218	mDNS Detection (Remote Network)
Info	N/A	10114	ICMP Timestamp Request Remote Date Disclosure
Info	N/A	45590	Common Platform Enumeration (CPE)
Info	N/A	54615	Device Type

Info	N/A	19506	Nessus Scan Information
Info	N/A	10335	Nessus TCP scanner
Info	N/A	11936	OS Identification
Info	N/A	117886	OS Security Patch Assessment Not Available
Info	N/A	70657	SSH Algorithms and Languages Supported
Info	N/A	149334	SSH Password Authentication Accepted
Info	N/A	10881	SSH Protocol Versions Supported
Info	N/A	153588	SSH SHA-1 HMAC Algorithms Enabled
Info	N/A	10267	SSH Server Type and Version Information
Info	N/A	22964	Service Detection
Info	N/A	25220	TCP/IP Timestamps Supported
Info	N/A	110723	Target Credential Status by Authentication Protocol - No Credentials Provided
Info	N/A	10287	Traceroute Information

*Table 3 - Nessus Results of Charging Infrastructure*

\* indicates the v3.0 score was not available; the v2.0 score is shown.

In this Nessus scan for Charging Infrastructures for Electrical Vehicles we collect in total 17 results. One of the totals belongs to a Medium category and the rest are Info.

The Medium severity is about the mDNS Detection which is possible to obtain information about the remote host. This remote service is able to understand the Bonjour or also known as ZeroConf or mDNS protocol. To understand the Bonjour protocol, Bonjour uses industry standard IP protocols to enable the automated discovery of devices and services on a local network [50].

Subsequently, the mDNS protocol allows anybody to learn about the distant host's operating system type and specific version, hostname, and list of services it is running.

This plugin tries to find mDNS used by hosts that are not on the network segment where Nessus is located.



### 5.2.3 *Exploitation Phase of the Charging Infrastructure*

In this case, we were not able to identify any vulnerability in the system, but some possible scenarios that could work in some cases are [51]:

- Spoofing or Man in the Middle Attack (MitM)
- Abuse the probing phase

In the former, the MitM attack can be performed between the client and the real server. That means the attacker might be able to obtain sensitive files or even credentials.

In the latter case, when a responder wants to start or change the connectivity, it checks the local network to see whether there is any resource with the name he intends to utilize. If the response contains the questioned record, the probing host should use a different name. If 15 conflicts occur within 10 seconds, the host must wait at least five seconds before making another attempt. Furthermore, if the host cannot discover an unused name within one minute, it reports an error to the user.

## 6. Results

As we discussed earlier, we analyzed two different networks in terms of their security and integrity. In these two networks we tried to find vulnerabilities that could damage the system so that the attacker could gain access to the network.

In the case of the SDN network we found that there was a vulnerability that led us to gain root of the system.

In any case we also had Nessus where the information we got from it was more important than we expected, the results were extremely important since they gave us information about vulnerabilities that the system could be attacked, and those in total were 7 of which 2 were critical and one of them gave us access to the system which was Docker.

In this particular case, there was nothing that the attacker could get from the system but in a hypothetical case where behind this SDN network there was an IoT network, where the IoT network provides more direct control over routing, analyzes the network traffic, and effectively manages the network, so in this kind of network the attacker could cause damage by obtaining this data and perhaps could modify or alter the data to cause disruptions in that network.

In the case of the second network where it was the Charging Infrastructure, although we could not get any access to the network, Nessus gave us a vulnerability that could potentially damage the network.

This vulnerability is mDNS and was the only one found in the system. Although we could not gain access this vulnerability is affected in 2 attacks.

- Man in the Middle Attack
- Abusing the Probing phase

Both attacks were discussed in detail in chapter 5.2.3 and are extremely important because each one individually can affect the system and its operation.

## 7. Conclusion

The rapid progression of the technology and Internet of Things changed the way that we prioritize things today. As time passes Critical Infrastructure becomes more and more important to a society, so it must be safe from either cyber attacks or natural disasters. Due to these attacks, it is important to do thorough checks and penetration testing on such infrastructures in order to cover their vulnerable points.

In this thesis we analyzed two kinds of systems. Firstly the SDN system and secondly the Electrical Vehicle charging infrastructure. In the SDN system penetration testing was done with the following procedure:

1. Reconnaissance (Nmap)
2. Vulnerability Identification (Nessus)
3. Exploitation (Penetration testing)

For the reconnaissance phase, we used the nmap tool to map the network and get all the possible open ports that the network has. We found in total 13 open ports with the docker being the most important of them. After the mapping process, we continue with the vulnerability analysis which gave us an analysis of all the networks and the possible vulnerabilities that it has. Lastly, we continued with the exploitation which we were able to get access as a root to the docker port.

In the Charging Infrastructure for Electrical Vehicles we started the same process and the results were fewer than the SDN. We had fewer open ports to the network and that didn't give us a lot of options to search for in the network. The only vulnerability that we were able to identify was mDNS which we weren't able to get access to.

In any case, it is necessary to secure these infrastructures in every way from every perspective so that there is a steady flow of operation of these infrastructures.

## 8. Future Work

Future extensions of this work could include research into the Critical Infrastructure part of other networks, as well as testing new ways of tackling vulnerabilities. For example, we could create new ways of protecting some infrastructures and there could be a detailed comparison between existing methods compared to the new methods. That way we can analyse which method works best on each infrastructure so that there is an extra layer of security.

In addition, we could build more secure environments for these infrastructures by

1. Blocking some ports that are not necessarily useful to the public. In this way, it will be difficult for the attacker to find an access point to the system so that the attacker can gain access to it.
2. Even a renewed form of permissions so that only the necessary people have access to the infrastructure. When multiple people have access to infrastructure this results in a greater chance that a third party may be able to intercept either passwords or a

handshake on the network and gain access to it. The reasoning behind only allowing access to the people who need it is that there will be less chance of error when an attack on the system is about to happen.

3. Creating ways where every move made on infrastructures is monitored. By recording all movements in a database, we can check at any time what went wrong in the system to restore it to its original state.
4. Build a defensive environment. Creating an environment around critical infrastructure to be ready to deal with any attack will help protect that infrastructure against possible future attacks.

In general, due to the increased utility of Critical Infrastructure in today's world, the need to create state-of-the-art ways of security, either physical or cyber, will help prevent future attacks where that can affect an entire network, as well as the daily lives of thousands of individuals.

Since the citizens of a country rely primarily on Critical Infrastructure, we need to eliminate any risks so that there is a smooth functioning of the infrastructures.

## 9. Bibliography

- [1] CISA GOV. (2019). *IDENTIFYING CRITICAL INFRASTRUCTURE DURING COVID-19* [Online]. Available: <https://www.cisa.gov/identifying-critical-infrastructure-during-covid-19>
- [2] A. Cantelli-Forti, A. Capria, A. L. Saverino, F. Berizzi, D. Adami, C. Callegari, “Critical infrastructure protection system design based on SCOUT multitech security system for interconnected space control ground stations”, *International Journal of Critical Infrastructure Protection*, vol. 32, pp. 100407, Nov. 2021.
- [3] Y. K. Dwivedi, D. L. Hughes, C. Coombs, I. Costantiou, Y. Duan, et al., “Impact of COVID-19 pandemic on information management research and practice: Transforming education, work and life”, *International Journal of Information Management*, vol. 55, pp. 102211, Sept. 2020.
- [4] A. Kaye, C. N. Okeagu, Alex. D. Pham, R. A. Silva, J. J. Hurley, et al.,” Economic impact of COVID-19 pandemic on healthcare facilities and systems: International perspectives”, *Best Practice & Research. Clinical Anaesthesiology*, vol. 35, pp. 293-306, Oct. 2021.
- [5] Nao.Org.uk. (Oct. 2017). *Investigation: WannaCry cyber attack and the NHS* [Online]. Available: <https://www.nao.org.uk/reports/investigation-wannacry-cyber-attack-and-the-nhs/>
- [6] “WannaCry ransomware attacks cost the NHS £92m”, *Computer Fraud & Security*, Vol. 2018, pp. 1-3, Aug. 2018.
- [7] weforum.org. (Feb. 2020). *A three-step path to securing critical infrastructure* [Online]. Available: <https://www.weforum.org/agenda/2020/02/a-3-step-path-to-securing-critical-infrastructure/>
- [8] A Guide to Critical Infrastructure Security and Resilience, Cybersecurity & infrastructure security agency, Nov. 2019. Available: <https://www.cisa.gov/sites/default/files/publications/Guide-Critical-Infrastructure-Security-Resilience-110819-508v2.pdf>
- [9] Fortinet.com. *Critical Infrastructure Protection (CIP)* [Online]. Available: <https://www.fortinet.com/resources/cyberglossary/critical-infrastructure-protection>
- [10] Supplemental Tool: Executing A Critical Infrastructure Risk Management Approach, Cybersecurity & infrastructure security agency, 2013 ,Available: <https://www.cisa.gov/sites/default/files/publications/NIPP-2013-Supplement-Executing-a-CI-Risk-Mgmt-Approach-508.pdf>
- [11] L. A. Maglaras, K.H. Kim, H. Janicke, M. A. Ferrag, S. Rallis, et al., “Cyber security of critical infrastructures”, *ICT Express*, vol. 4, pp. 42-45, Nov. 2018.
- [12] G. Yadav, P. Kolin, “Architecture and security of SCADA systems: A review”, *International Journal of Critical Infrastructure Protection*, vol. 34, pp. 100433, Jun. 2021.
- [13] N. Kaushik. (April 2012). *Difference Between PLC and RTU* [Online]. Available: <http://www.differencebetween.net/technology/industrial/difference-between-plc-and-rtu/>

- [15] D. H. Ryu, H. Kim, K. Um, “Reducing security vulnerabilities for critical infrastructure”, *Journal of Loss Prevention in the Process Industries*, vol. 22, pp. 1020-1024, Aug. 2009.
- [16] D. Galar, U. Kumar, “Chapter 4 - Data and Information Fusion From Disparate Asset Management Sources”, in *Book eMaintenance*, D. Galar, U. Kumar, Academic Press, Jan. 2017, pp. 179-234.
- [17] E. Staff. (Aug. 2017). *What is a Control Loop ? | Components of Control Loop* [Online]. Available: <https://instrumentationtools.com/what-is-control-loop/>
- [18] S. McLaughlin, D. Podkuiko, S. Miadzevzhanka, A. Delozier, P. McDaniel, “Multi-vendor penetration testing in the advanced metering infrastructure”, in *Conf. Twenty-Sixth Annual Computer Security Applications, ACSAC 2010*, Austin, Texas, USA, pp. 107-116.
- [19] en.wikipedia.org. *Electrical Grid* [Online]. Available: [https://en.wikipedia.org/wiki/Electrical\\_grid](https://en.wikipedia.org/wiki/Electrical_grid)
- [20] S. Wallace, X. Zhao, D. Nguyen, K. -T. Lu,” Chapter 17 - Big Data Analytics on a Smart Grid: Mining PMU Data for Event and Anomaly Detection” in *Book Big Data*, R. Buyya, R. Calheiros, A. Dastjerdi, Morgan Kaufmann, Jan. 2016, pp. 417-429.
- [21] G. Steriopoulos, E. Vasilellis, G. Lykou, P. Kotzanikolaou, D. Gritzalis, *CRITICAL INFRASTRUCTURE PROTECTION TOOLS: CLASSIFICATION AND COMPARISON*, 2016, Available: <https://infosec.aueb.gr/Publications/CIP-2016%20CIP%20Tools.pdf>
- [22] C. Koukoumialos, G. Stergiopoulos, *An Overview of Critical Infrastructure Analysis Software*, 2015, Available: <https://www.infosec.aueb.gr/Publications/2015-NOV-Poster%20CI%20Analysis%20Software.pdf>
- [23] dhs.gov. (USA 2009). *National Infrastructure Protection Plan, US Dept. of Homeland Security* [Online]. Available: [https://www.dhs.gov/xlibrary/assets/NIPP\\_Plan.pdf](https://www.dhs.gov/xlibrary/assets/NIPP_Plan.pdf)
- [24] M. Ouyang. “Review on modeling and simulation of interdependent critical infrastructure systems”, *Reliability Engineering & System Safety*, vol. 121, pp. 43-60, Jan. 2014.
- [25] D. Pliatsios, P. Sarigiannidis, T. Lagkas, A. G. Sarigiannidis, “A Survey on SCADA Systems: Secure Protocols, Incidents, Threats and Tactics”, *IEEE Communications Surveys & Tutorials*, vol. 22, pp. 1942-1976, 2020.
- [26] P. Radoglou Grammatikis, P. Sarigiannidis, G. Efstathopoulos, P. Karipidis, A. Sarigiannidis, “DIDEROT: an intrusion detection and prevention system for DNP3-based SCADA systems”, in *Conf. ARES 2020: The 15th International Conference on Availability, Reliability and Security*, May 2020, pp. 1-8.
- [27] M. P. Jarabo-Amores, M. Rosa-Zurera, D. Mata-Moya, A. Capria, et al., “Distributed Physical Sensors Network for the Protection of Critical Infrastructures Against Physical Attacks”, in *Conf. SPECIAL SESSION ON DATA COMMUNICATION FOR CRITICAL INFRASTRUCTURES*, Jan. 2016, pp. 139-150.
- [28] Cipher. (Feb. 2018). *A Complete Guide to the Phases of Penetration Testing* [Online]. Available: <https://cipher.com/blog/a-complete-guide-to-the-phases-of-penetration-testing/>

- [29] Eccouncil.org. *Learn About the Five Penetration Testing Phases | EC-Council* [Online]. Available: <https://www.eccouncil.org/cybersecurity-exchange/penetration-testing/penetration-testing-phases/>
- [30] S. Nahum. (Jan. 2021). *Penetration Testing Steps: Anatomy of a Successful Pentest* [Online]. Available: <https://www.securecoding.com/blog/penetration-testing-steps/>
- [31] Coresentinel.com. (June 2017). *The Difference Between White, Black, and Grey Box Penetration Testing* [Online]. Available: <https://www.coresentinel.com/black-box-vs-white-box-testing/>
- [32] P. Midian, “Perspectives on Penetration Testing — Black Box vs. White Box”, *Network Security*, Vol. 2002, pp. 10-12, Aug. 2002.
- [33] Rajkumar. (Jan. 2019). *18 Best Penetration Testing Tools (Free and Paid) for 2022* [Online]. Available: <https://www.softwaretestingmaterial.com/penetration-testing-tools/>
- [34] Simplilearn.com. (Nov. 2021). *What is Metasploit: Overview, Framework, and How is it Used | Simplilearn* [Online]. Available: <https://www.simplilearn.com/what-is-metasploit-article>
- [35] w3af.org. *Features | w3af - Open Source Web Application Security Scanner* [Online]. Available: <https://w3af.org/features>
- [36] docs.tenable. *Benefits and Limitations (Nessus Agents)* [Online]. Available: <https://docs.tenable.com/nessusagent/Content/BenefitsAndLimitations.htm>
- [37] hcltechsw. *AppScan Dynamic Application Security Testing (DAST) - HCL Software* [Online]. Available: <https://www.hcltechsw.com/appscan/offerings/standard>
- [38] portswigger.net. *Features - Burp Suite Professional* [Online]. Available: <https://portswigger.net/burp/pro/features>
- [39] kali.org. *hydra | Kali Linux Tools* [Online]. Available: <https://www.kali.org/tools/hydra/>
- [40] kali.org. *sqlmap | Kali Linux Tools* [Online]. Available: <https://www.kali.org/tools/sqlmap/>
- [41] wireshark.org. *Wireshark · About* [Online]. Available: <https://www.wireshark.org/about.html>
- [42] esecforte. *Netsparker professional- Web application security Scanner* [Online]. Available: <https://www.esecforte.com/products/netsparker-web-application-security-scanner/>
- [43] rapid7. *Vulnerability Scanning Software Features* [Online]. Available: <https://www.rapid7.com/products/nexpose/features/>
- [44] coresecurity. *Core Impact | Penetration Testing Software | Core Security* [Online]. Available: <https://www.coresecurity.com/products/core-impact#features>
- [45] J. English. *SDN controller (software-defined networking controller)* [Online]. Available: <https://www.techtarget.com/searchnetworking/definition/SDN-controller-software-defined-networking-controller>
- [46] M. Roomi. (June 2021). *5 Advantages and Disadvantages of SDN | Drawbacks & Benefits of SDN* [Online]. Available: <https://www.hitechwhizz.com/2021/06/5-advantages-and-disadvantages-drawbacks-benefits-of-sdn.html>

- [47] J. Borges, C. Ioakimidis, P. Ferrão, “Fast charging stations for electric vehicles infrastructure”, in *Conf. WIT Transactions on Ecology and the Environment*, Mar. 2010, pp. 275-284.
- [48] J. A. Lopes, F. Soares, P. Almeida, M. Moreira da Silva, “Smart Charging Strategies for Electric Vehicles: Enhancing Grid Performance and Maximizing the Use of Variable Renewable Energy Resources”, in *Conf. EVS24 International Battery, Hybrid and Fuel Cell Electric Vehicle Symposium*, Jan. 2009.
- [49] N. Bhusal, M. Gautam, M. Benidris, *Cybersecurity of Electric Vehicle Smart Charging Management Systems*, Aug. 2020.
- [50] developer.apple. *About Bonjour* [Online]. Available: <https://developer.apple.com/library/archive/documentation/Cocoa/Conceptual/NetServices/Introduction.html>
- [51] book.hacktricks. *5353/UDP Multicast DNS (mDNS) and DNS-SD - HackTricks* [Online]. Available: <https://book.hacktricks.xyz/network-services-pentesting/5353-udp-multicast-dns-mdns>