
This is the **published version** of the bachelor thesis:

Carretero Canut, Adrià; Franco Puntos, Daniel, dir. Cloud-featured micro segmentation solution for endpoints. 2023. (958 Enginyeria Informàtica)

This version is available at <https://ddd.uab.cat/record/272788>

under the terms of the  license

Cloud-featured micro segmentation solution for endpoints

Carretero Canut, Adrià

Abstract— Multinational companies are succulent targets for hackers who want to make business out of their attacks, or even for state-sponsored threat actors such as Cozy Bear (APT29) or the Lazarus Group (APT38), who either try to steal sensitive information or spoil a company's services or reputation. To achieve so, hackers must move laterally within the company's servers to collect new information, gain privileges or have access to new assets in order to achieve their ultimate objective. Network segmentation has typically being applied to harder Lateral Movement attacks, but a new approach based on modifying an endpoint's local firewall to prevent these actions from happening has raised lately. This thesis aims to contribute to the Cybersecurity Open-Source community by creating a simple, modular micro segmentation framework that allows IT administrators to prevent cyberattacks by deploying custom rules on Windows endpoints.

Keywords— Network-level security, Cloud, Micro segmentation, Cybersecurity, Endpoint protection, Lateral Movement, PowerShell

Resumen— Las compañías multinacionales son un succulento objetivo para los hackers que pretenden perpetrar ataques cibernéticos con tal de conseguir un beneficio monetario, o para los grupos esponsorizados por ciertos gobiernos como “Cozy Bear” (APT29) o “Lazarous Group” (APT38), cuyo propósito es robar información sensible o dañar los servicios o la reputación de la compañía. Para conseguirlo, los hackers deben moverse lateralmente por los servidores de la compañía para recolectar información, escalar privilegios o tener acceso a nuevos dispositivos para lograr su objetivo. La segmentación de la red ha sido típicamente aplicada para impedir los movimientos laterales, pero un nuevo enfoque centrado en la modificación de los firewalls locales de los dispositivos de negocio ha ido creciendo últimamente. Este trabajo tiene como finalidad contribuir a la comunidad Open-Source mediante la creación de un framework para la microsegmentación de dispositivos simple y modular que permita a los administradores prevenir ciberataques mediante la implementación de reglas de firewall en dispositivos Windows.

Palabras clave— Seguridad a nivel de red, Cloud, Microsegmentación, Ciberseguridad, Protección de dispositivos, Movimiento Lateral, PowerShell

1 INTRODUCTION

CYBERSECURITY has become one of the most thriving, yet threatening fields of the IT world during these recent years. Every day individuals are robbed, politicians spied, and companies hacked. Cybersecurity has even taken a crucial role in the war between Russia and Ukraine. So proof it the latest studies on cyber threats which, at the time of writing, are accountant for the rising

malwares and attacks around the world. Between March 2021 and February 2022, 153 million new malware samples were created, which represent a 5% increase on the previous year, nearly 50% of business PCs got infected and 53% of business PCs got reinfected (therefore had been infected previously), and 86.2% of companies surveyed by “CyberEdge Group” in 2021 were affected by a successful attack [1]. Nevertheless, the IT community is evolving to prevent these attacks from happening and keeping individual and company assets safe.

2 OBJECTIVES

The aim of this project is to secure a company's Windows endpoints by allowing administrators to set new rules. This is achieved by using MURO, a modular, cloud-oriented

• E-mail of contact: adrian.carreteroc@autonoma.cat
 • Specialization: Computer engineering
 • Thesis director: Daniel Franco Puentes (Computer Architecture and Operating Systems Dpt)
 • Curs 2022/23

tool. Some of the features that ought to be highlighted are:

O1. Centralized control: having a unique, sole point of control reduces complexity, and eases the control of the micro segmentation strategy deployment.

O2. Usage of PowerShell capabilities: MURO leverages Windows' native tool "PowerShell" with its default commands, to deploy local firewall rules across the infrastructure, preventing the administrators from installing additional tools or frameworks.

O3. Cloud-oriented tool: its modular design allows developers to implement a "rule-synchronization" module, allowing IT administrators to share their rules through a repository hosted in the cloud, providing more elasticity when it comes to performance. Just as if it was a "market-place".

O4. Zero trust approach: as in 2022, 84% of companies have decided to adopt zero trust strategies [2], which are meant to continuously validate every stage of digital interaction even if the activity is found inside the companies perimeter [3].

Additionally, in this thesis, some personal goals to be accomplished are:

O1. To learn about Active Directory (AD): nowadays, Windows' AD is "de facto" directory service, or "hierarchical structure to store information about objects on the network" [4] in large companies. Positions such as network engineer or cybersecurity engineer (due to its relevance within a company's infrastructure) require technicians to have, at least, basic notions on AD.

O2. To learn more about system administration: approximately 1.4 billion devices around the globe [5] run Windows 10 or Windows 11 monthly. Furthermore, AD has historically been the solution implemented to manage large directories, and Windows Server management skills is often demanded.

O3. To learn about network security: to make it more difficult for attackers to move across a company's servers, the network is usually segmented [6]. A new approach has been made lately, and micro segmentation for endpoint has grown to hinder even more the possibility of lateral movements from happening, preventing hackers from moving from one server to another one in an attempt to achieve their ultimate goal.

3 PLANNING

The aim of this project is to develop a modular tool that allow administrators to set new rules on a company's Windows endpoints to enhance security. The project can be divided into the following stages.

- S1. Emulate a multinational company's infrastructure.
- S2. To code the rule edition program.
- S3. To develop a Graphical User Interface (GUI).
- S4. Provide a secure deployment plan.

To test the tool and adjust it to a real environment, a multinational company's architecture was implemented using VMware Workstation Pro (S1). The second stage, S2, consists on developing the scripts that gather the information from the Active Directory and deploy the rules according to it. Next, S3 pretends to implement a GUI so the tool is more friendly to the users. Lastly, stage 4 (S4) intends to provide a plan to securely implement MURO in a "real-world" business, along some additional recommendations.

Fig. 15 is a Gantt diagram that shows, in detail, how this project was planned. It can be found in the Appendix section.

4 STATE OF THE ART

Since some years ago, network segmentation has been used by companies from all around the world to segment their networks in an effort to reduce the attack surface, improve breach containment and strengthen regulatory compliance. However, the "new world order" seems to have started. The cybersecurity company Illumio has developed a game changer product called Illumio Edge [7], which allow companies to set specific rules for concrete or a group of local firewalls in workstations from a central server. This product is meant to foment Zero Trust policies, which is defined by the idea that "no one is blindly trusted and allowed to access company assets until they have been validated as legitimate and authorized" [14].

After some research was conducted, it is been concluded that a few providers, not to say none, offer a solution comparable to Illumio Edge. While there are many alternatives to Illumio Core [15], which focuses on data centers and the cloud, Illumio Edge focuses on endpoints, which serves as a first measure to prevent lateral movement or ransomware coming from infected USBs or files to be successfully carried out.

5 METHODOLOGY

After reviewing multiple project management methodologies to control MURO, such as the Waterfall methodology [8], the "Agile" technique [9] was chosen. "Agile" encourages the continuous development of the project while permitting some flexibility when it comes to adding new features throughout the project, as short deadlines are set [10]. The four stages defined in section 3 are sequential. However, a testing stage was defined for each part of the software, adding an additional layer of validation that prevented the "Testing and Validation" stage from taking too long.

Next, each phase of the project is explained following the order listed in section 3.

- S1. Emulate a multinational company's infrastructure.

5.1 Building the infrastructure

One of the main features of this tool is that it was thought to be used with infrastructures that include a Windows Active Directory. Large multinational enterprises typically use this technology to manage the access of their employees not only to computers, but tools such as Outlook, Sharepoint... Active Directory allows IT Administrators to organize users and computers in groups, easing group policy (GPO) deployment and controlling access, among others.

Because of that, the first step was focused on emulating a multinational company's infrastructure where MURO (the tool) can be tested. Just as the name says, a "multinational company" is based in "multiple nations", and therefore different physical networks. Moreover, large enterprises usually have an "on-premise" network and then other services in the cloud which are out of network "trust".

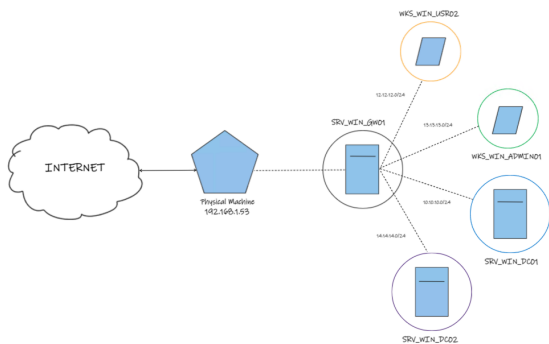


Fig. 1: iCorp's (fictitious multinational company) infrastructure developed in my personal lab.

As shown in Fig. 1, the infrastructure is composed by the following elements:

- **1 gateway** – machine "SRV_WIN_GW01" is a Windows 2019 server used as a gateway to the "outer world" and router (since each computer is found within an independent network and, to contact others, traffic must be redirected by this "server gateway"). The company's endpoints access the "real world" through a set of gateways, which have a firewall installed on them. Server GW01 is meant to emulate that device.
- **2 domain controllers** – these domain controllers (DC) are Windows 2019 Servers. While "SRV_WIN_DC01" responds to domain "icorp.local", DC "SRV_WIN_DC02" is responsible for the child domain "am.icorp.local". Just as with large multinational companies, often a forest must have multiple domains, including parent and child domains to differentiate between time zones.
- **2 workstations** – finally, two Windows 10 machines were deployed. "WKS_WIN_ADMIN01" corresponds to the administrator's computer belonging to domain "ICORP\" and is being used to develop the core scripts. Unlike the previous workstation, "WKS_UBU_USR02" belongs to domain "AM\" and, in addition to add more information when enumerating

assets, it will be used as the "target machine" when deploying firewall rules.

Building the infrastructure was complex since each workstation is associated to a domain, is isolated in its own network and none of them have a direct access to internet, meaning that a DNS had to be configured specifically, among others. Adding all these up made quite difficult to deploy the infrastructure.

Fig. 2 shows all computers enrolled in the Active Directory, while Fig. 3 shows the child domain controller's Server Manager panel:

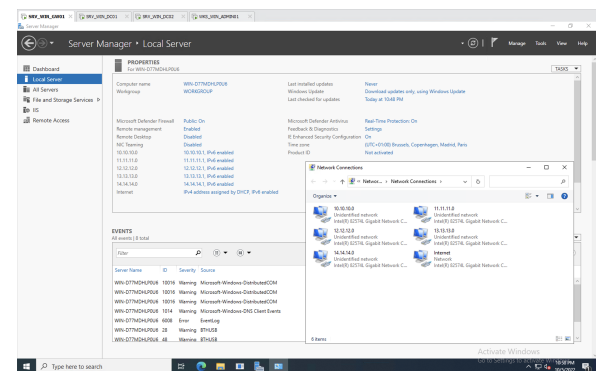


Fig. 2: Server "WIN.SRV_GW01" Server Manager's panel

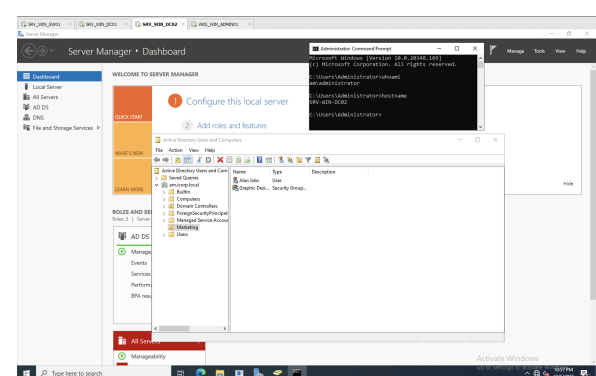


Fig. 3: Domain controller SRV_WIN_DC02 showing AM domain's users

Fig. 4 proves the correct implementation of the Active Directory as user "bgates" belongs to domain "ICORP" and the username shown is "ICORP\bgates". The different users enrolled in the Active Directory were used to log in different computers, successfully, which served to validate the correct implementation of the AD.

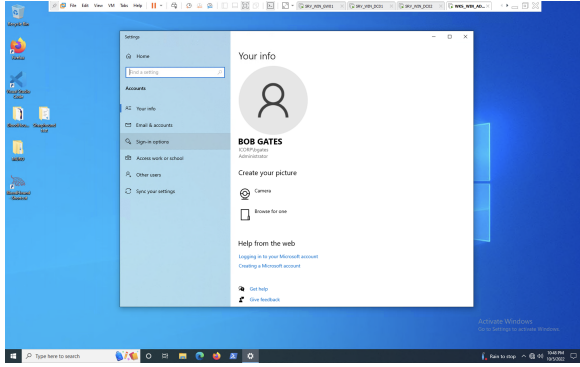


Fig. 4: User Bob Gates' profile from domain "ICORP" once logged in workstation WKS_WIN_ADMIN01

5.2 Scripting

This second stage of the project consisted on creating the scripts that make the endpoint micro-segmentation possible.

"MURO" is meant to edit Windows endpoints' local firewall, and so it was decided to leverage PowerShell's capabilities by coding scripts built on the .NET Common Language Runtime (CLR) [16], which can also run in MacOS and Linux operating systems, providing certain flexibility for future implementations. What is more, PowerShell was chosen as it allows to follow a "Living off the Land" (LotL) approach [11]. This approach consists on the usage of legitimate software and functions available to reduce the amount of extra tools or packages that might have to be installed in order to use the tool. Lastly, it includes native cmdlets that allow accomplishing MURO's objectives, which would be more complex with other solutions such as Python. Two scripts form MURO:

1. **AD Enumerator:** once a domain is given, it extracts all the information regarding computers, domains, forests, groups, organizational units (OUs) and users from that domain and all its child domains, storing the information in separate files in JSON format.
2. **Firewall Editor:** it edits the local firewall for the specified assets by completing the actions specified by the user.

5.2.1 AD Enumerator (.ps1)

Usually, the "AD Enumerator" script would be run first to update the information from the Active Directory. Depending on the company's size, much information is retrieved from the Domain Controllers, so it is recommended to run the script every day at a certain time.

Once it finishes, all the information is stored in JSON files which are accessed by the Graphical User Interface (explained later) to provide the user with the information contained within the AD of the company. Fig. 5 is the result of a test conducted to check the correct implementation.

```
{
  "wks": [
    {
      "Name": "SRV-WIN-DC01",
      "DNSHostName": "SRV-WIN-DC01.icorp.local",
      "Enabled": true,
      "ObjectClass": "computer",
      "OperatingSystem": "Windows Server 2022 Standard"
    },
    {
      "Name": "WKS-WIN-ADMIN01",
      "DNSHostName": "WKS-WIN-ADMIN01.icorp.local",
      "Enabled": true,
      "ObjectClass": "computer",
      "OperatingSystem": "Windows 10 Education N"
    },
    {
      "Name": "WKS-UBU-USR01",
      "DNSHostName": null,
      "Enabled": true,
      "ObjectClass": "computer",
      "OperatingSystem": null
    },
    {
      "Name": "SRV-WIN-DC02",
      "DNSHostName": "SRV-WIN-DC02.am.icorp.local",
      "Enabled": true,
      "ObjectClass": "computer",
      "OperatingSystem": "Windows Server 2022 Standard"
    }
  ],
}
```

Fig. 5: Information about the computers belonging to iCorp's AD extracted by the "AD Enumerator" script

5.2.2 Firewall Editor (.ps1)

The "Firewall Editor" script is in charge of creating, setting or getting rules from the local firewall of the specified endpoints. It does so by using PowerShell's cmdlet "XXX-NetFirewallRule" (included in the NetSecurity module) [17], where 'XXX' is defined by the user as there are multiple cmdlets that interact with a machine's local firewall. Currently, options "New", "Set" and "Get" are available in script "fwEditor.ps1".

Many tools in the market use its own syntax to function, and MURO is no different. Users shall follow this syntax to construct rules. As mentioned previously, MURO leverages native cmdlet "XXX-NetFirewallRule" to create rules. This cmdlet has its own parameters, and MURO "maps" this parameters to the user's input. A basic structure is followed when defining a rule:

<MURO-param1>:<value1>;<MURO-param2>:<value2>;...;<MURO-paramN>:<valueN>

An example of a command would be the following:

mode:new;action:block;port:8888;protocol:TCP;name:Test_Port8888;dispname:Test_Port8888;cust_targ:WKS-WIN-USR02.am.icorp.local

MURO's documentation (which can be found in its repository) specifies which parameters are implemented currently, and its corresponding "nickname" in MURO's syntax.

After receiving the user's input, MURO parses the different fields and creates the "XXX-NetFirewallRule" command dynamically by appending the parameters and its corresponding value to a string. Yet, during a test, it was seen that cmdlet "xxx-NetFirewallRule" did not

use parameter “-RemoteComputer” to set rules in remote computers, but to define rules to allow or block traffic from certain remote computers. In other words, an additional cmdlet must be used to specify the target computer where the “XXX-NetFirewallRule” is run.

To solve that issue, it was decided to run the command in the target computer through the Windows Remote Management (WinRM) capability [18]. Cmdlet “Invoke-Command” allows to execute commands in remote computers through parameter “-ComputerName” if credentials used to access the target computer are valid (parameter “-Credential” is used) [19]. After some tests, the Firewall Editor script was edited to make use of cmdlet “Invoke-Command” and run the “XXX-NetFirewallRule” cmdlet in the specified target computer. Nevertheless, a new issue arose: the usage of credentials.

As mentioned earlier, in order to run a command in a remote computer, valid credentials must be provided in cmdlet “Invoke-Command”. This credentials (username and password) are sent as a “PSCredential object”, which is not a string. Additionally, it has been said that, when generating the command dynamically, parameters (strings) are attached to the final command, and so are credentials. As credentials become part of the command (an string), PowerShell does not interpret them as an object, but as a string, which are not recognized as valid credentials by the target computer, and therefore the firewall rule cannot be deployed.

To prevent sending credentials through the net, a special “super user” was created. In Windows systems, Local Administrators are able to run commands remotely without the need of introducing their credentials. Following this new approach, two actions were taken:

1. Create a special user
2. Creating a new GPO (Group Policy Object)

As mentioned earlier, local administrators in a target computer are able to run commands remotely without introducing their credentials. Thus, in order to deploy firewall rules in any endpoint within the Active Directory, a special user for MURO with local administrator privileges in all machines had to be created. This special user, named ‘MURO_SS’, is the only one allowed to create firewall rules in any endpoint across the Active Directory (alongside the Domain Administrator). Centralizing the local firewall rule creation provides an additional layer of security, as all firewall rules that are created under other user accounts can be considered malicious activity.

Secondly, a “Local Administrators” group (where ‘MURO_SS’ is member of) and a Group Policy Object (GPO) setting the “Local Administrators” group member as local administrators in any workstation within the AD were created. A tutorial [20] was followed to create such GPO.

After completing both actions, the “-Credential” parameter was no longer included in the commands that are generated dynamically, allowing the script to deploy

firewall rules successfully (test result in Fig. 6).

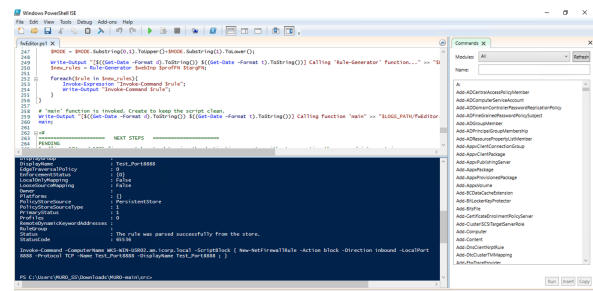


Fig. 6: A firewall rule is created successfully through the command line

Finally, some additional features that MURO includes are listed below:

- **Defining profiles:** “profiles”, JSON files following a preset structure (Fig. 7), can be created to set multiple new rules in one or more endpoints. In combination with option “Targets”, it can be used to onboard thousands of devices or to patch thousands when, for example, a vulnerability related to SSH (port 22) is published. Fig. 8 is an example of a “Targets” file



Fig. 7: Example of a “Profile” file

- **Specifying the targets:** just as profiles, “targets” allows the user to specify a group of devices to which one or multiple rules are applied.
- **Run custom rules through the GUI:** the “Profiles” and “Targets” flags are optional, so the tool can also be used to run individual commands in combination with the Active Directory information extracted by the “AD Enumeration” script.

5.3 Creating the GUI

Although the aforementioned scripts were also designed to be used through the console, deploying different rules

```

{
  "Inbound": [
    {
      "title": "IPs",
      "recipients": ["13.13.13.129", "12.12.12.129"]
    },
    {
      "title": "Computers",
      "recipients": ["WKS-WIN-ADMIN01", "WKS-WIN-USR02"]
    }
  ],
  "Outbound": [
    {
      "title": "Groups",
      "recipients": ["CISOC"]
    },
    {
      "title": "OUs",
      "recipients": ["IT"]
    }
  ]
}

```

Fig. 8: Example of a “Targets” file

to multiple endpoints by using a Graphical User Interface (GUI) is possible. MURO provides a website that shows all the content extracted by the “AD Enumeration” script from the AD and allows users to submit their commands by using that information.

The website’s front end, which loads the content from the JSON files storing the AD’s information and submits the command created by the user, was created using the Vue.js framework, while the back end, which transfers the user’s command to the Firewall Editor script and executes it, was created using the Java Spring Framework.

5.3.1 Front end

At the beginning, the front end part was coded using raw HTML. However, due to many reasons such as the integration with the back end and the aesthetics, among others, it was decided to use Vue.js. Since it was the first time that Vue.js was used, it was decided to use a Vue.js template from Vue.js’ official webpage [12]. Fig. 9 shows the original website that was modified to fulfill MURO’s requirements.

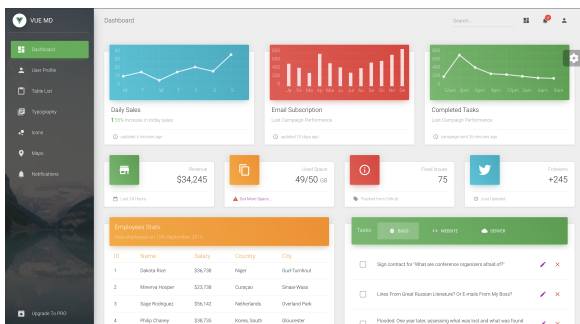


Fig. 9: Original website once a fresh, local server was up and running

The original website includes some features that do not fit in MURO’s concept, although might do in a future, such as the statistics panel or the increase of followers in Twitter. Moreover, the basic functionality of MURO is quite clear,

and led to three tasks that had to be fulfilled to accomplish a simple and easy to use interface:

1. **Remove all superfluous features:** As shown in figure 9, some features can be removed. In addition, there are some sections in the side bar that won’t be used, such as the “Map” or the “Typography” ones.
2. **Add an input bar:** this is the main feature of the website. By making use of it, the user is able to write the rule he/she wants to deploy in some of the company’s endpoints.
3. **Import and display information from JSON files:** one of MURO’s features is to enumerate the company’s AD and display the information so the IT administrator can make an efficient use of it. This information shall be displayed in an organised way, and using only those columns that are considered to be necessary.

After some days of editing several files of the template, the website looks as in Fig. 10:

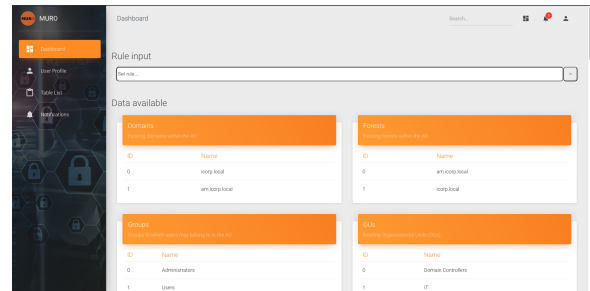


Fig. 10: MURO website’s Home page

Furthermore, not only code was deleted or commented out, but additional code was written to read the user’s input and send it to the back end side using HTTP requests. Fig. 10 does also show how data is imported correctly from the multiple JSON files, “passing” the test for this stage satisfactorily. After its completion, it was time for the back end.

5.3.2 Back end

The second part of the website was created using the Java Spring framework. In fact, website Spring Initializr [13] was used to create a frame that could be easily edited to receive the text from the front end and send it to the scripts stored in the server. Then, the Firewall Editor script uses the information received as its input, and runs to edit the target’s local firewall.

Fig. 11 shows the configuration used to create the back end’s frame.

No specific test was conducted for the back end part. After running it, no errors prompted, indicating that the implementation was correct. The correct functioning was tested later on during the “Final test” stage.

5.3.3 Final test

At this point, both the front end and the back end had been implemented, proving to work correctly as they were tested

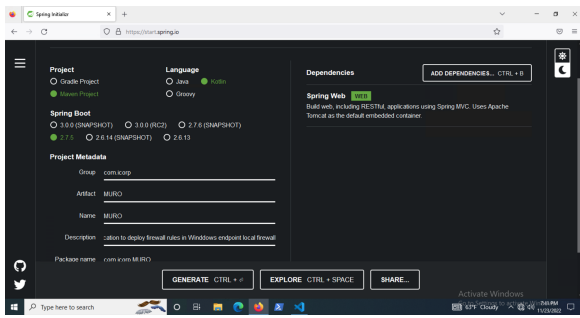


Fig. 11: Configuration for Spring Initializr to create back end template

successfully (independently). After that, it was time to test the entire application by assembling both parts. User MURO_SS was used as it has Local Administrator privileges in any computer within ICORP's infrastructure. Fig. 12 and Fig. 13 show how the payload was received by the back end, and the firewall rule created in the target machine.

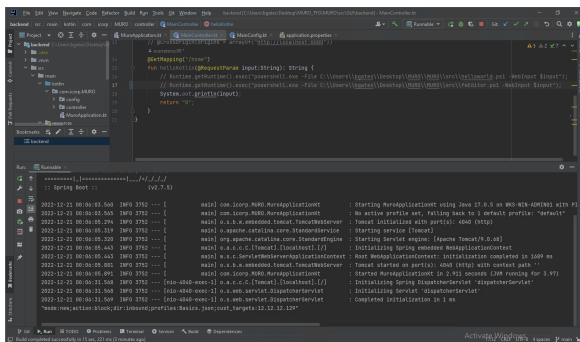


Fig. 12: Payload from frontend is printed in IntelliJ's terminal (backend editor)

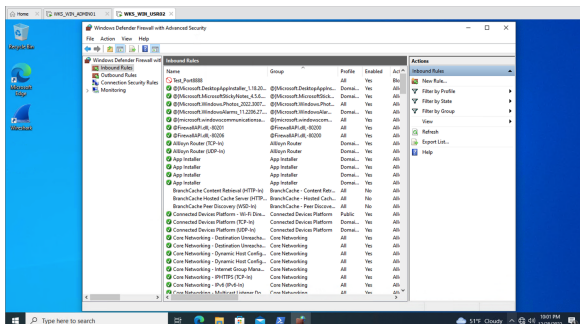


Fig. 13: Firewall rule created using GUI in workstation WKS-WIN-USR02

5.4 Secure integration to the company's infrastructure

To end this thesis with, a deployment model is proposed to ensure that MURO does not alter other assets' behaviour when installing it. Furthermore, since MURO is a cybersecurity-oriented tool and needs high privileges to function, it is crucial to integrate it in the company's infrastructure securely.

This section is divided into two sub sections: on-premise, where the server having MURO installed is found, and the cloud, where the global repository is found.

5.4.1 On-premise

“On-premise”, also known as “Trust”, is defined as the network whose assets are hosted on-site (servers, routers, workstations...) where all traffic is trusted. The frontier between the Internet and on-premise network is defined by the company's firewalls, where the multiple IT administrators set their own rules to protect the company's servers. It is recommended to own the server where MURO's main console is installed and host it on premise as all security considerations taken to date will also applied.

5.4.2 On-premise recommendation

Some additional recommendations to consider are:

1. **Usage of a container:** a container is a “standard unit of software that packages up code and all its dependencies so the application runs quickly and reliably from one computing environment to another” [21]. In this case, however, it is advised to deploy the application using a container to add an additional layer of security.
2. **Access List / Permissions:** not only permissions to access the container shall be applied, but also to the server hosting the container to hinder the access to hackers.
3. **Install EDR in hosting server:** an EDR (Endpoint Detection and Response) is a solution that monitors continuously the devices where its agent is installed to detect and respond to cyber threats intelligently [22]. It is one of the most advanced tools, not to say the most, when it comes to secure a company's assets, and it is therefore recommended to install an agent where the container is deployed.
4. **Event logs sent to SIEM:** a SIEM (Security Intelligence and Event Management) is a component aimed to receive logs from multiple sources and, according to the rules/use cases defined in it, offenses will trigger so security analysts can handle the incidence [23]. Events generated by the EDR are sent to the SIEM, and usually are enough to investigate security incidents, but due to the criticality of the running service, it is recommended to send some of the Microsoft Event Logs of the server hosting the container and the container its self to detect possible security incidents.
5. **DMZ implementation:** following the “Layered Security” principle [24], it is recommended to implement a DMZ (DeMilitarized Zone) where the server contacting the AWS instance is hosted. A DMZ is “a perimeter network that protects and adds an extra layer of security to an organization's internal local-area network from untrusted traffic” [25]. By placing the server there, HTTP/S and other legitimate traffic will be prevented from reaching server running the container, reducing the attack surface.

5.4.3 Cloud

Nowadays, multiple enterprise tools offer customers the opportunity to share their knowledge with others. MURO is no different, and it is intended to allow users from different companies to share “Profiles”, rules and others through its cloud-hosted, public website (in a future). The following section describes the ideal “marketplace’s” infrastructure, oriented to data protection and the customer’s security.

5.4.4 Cloud infrastructure

1. **Internet Gateway:** this component is “a horizontally scaled, redundant, and highly available VPC component that allows communication between your VPC and the internet”. [26] It is aimed to connect the EC2 instances hosting MURO’s website in the cloud with the internet to offer clients access to the firewall rules and profiles created by the community.
2. **Elastic Load Balancing (ELB):** just as regular load balancers, it is in charge of automatically distributing the incoming traffic across multiple targets (in this case, the EC2 instances hosting MURO) in one or more Availability Zones (AZ) [27]. This prevents a single EC2 instance from saturating and offering a slow, poor service to clients.
3. **EC2 instance:** these virtual machine servers provide scalable computing capacity in AWS’ cloud [28], which act as regular hosts running MURO’s GUI to offer firewall rules and profiles to the community. Multiple EC2 instances can be deployed to offer a comfortable service, and optimize the resource usage to reduce the cost. Additionally, EC2 instances can scale automatically, adapting its performance according to the workload.
4. **Elastic File System (EFS):** this AWS component provides serverless, fully elastic file storage [29]. EFS was chosen as the storage component to be used since 1) Multiple instances reading and writing simultaneously, 2) Stores data in and across multiple Availability Zones and 3) It automatically scales, growing and shrinking as files are added or removed.

Fig. 14 illustrates how MURO can be integrated securely in a multinational company’s infrastructure.

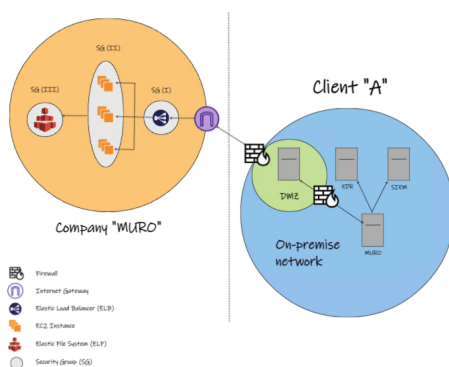


Fig. 14: On-premise recommended implementation and MURO’s cloud infrastructure

6 NEXT STEPS

MURO has just been created, and its modular implementation allows developers to create their own modules and integrate them into MURO. Furthermore, its implementation with PowerShell, a powerful and native Microsoft tool, allows developers to leverage Windows full capabilities to improve MURO.

Below, a list of next implementations that can be developed is proposed:

1. **AD data collection optimization:** nowadays, MURO runs the “AD Enumeration” script to collect multiple data from the company’s Active Directory and saves it into JSON files. For large enterprises, this solution might not be optimal as large storage will be needed, Active Directory updates constantly and some time might be needed to extract all data. Instead of pulling all the AD data, updating it with only the new changes might be feasible solution.
2. **Expand its functionality:** to January 2023, only 10 parameters are available in MURO. More can be added to provide more flexibility and allow IT administrators to create more complex rules to defend a company’s endpoints.
3. **AI implementation:** MURO allows to deploy “static” firewall rules based on the IT administrator’s criteria. Yet, what if MURO was capable of adding or removing firewall rules according to the usual traffic detected on a server? This would allow suspicious connections to be closed instantly!
4. **Expand to other OS:** right now, MURO only works with Windows devices. However, as mentioned in section 5.2, PowerShell runs in both MacOS and Linux, allowing developers to include a PowerShell module to deploy firewall rules in all three major operating systems.

7 CONCLUSIONS

As the number of cyberattacks increase daily, creating new solutions to hinder threat actors is necessary. To this date, January 2023, MURO is not “solid” enough to be implemented in a large multinational company as its GUI is not completed, available parameters can be extended, and its integration with the AD can be improved. Moreover, the cloud side has not been developed, and most important, enterprise solutions such as Illumio Edge have taken a big lead.

This thesis, however, has laid the foundation to create a modular tool that allows the community to prevent Lateral Movements and other red teaming techniques from happening. By following the next steps listed in the previous section, MURO can become an open source tool that can be implemented in small multinational companies or, at least, serve investigation purposes.

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my family, whose unconditional love and support has helped me out throughout these last five years.

Special thanks to my friend Marc Núñez, who guided me through the back end process of creation. I am also grateful to all my friends with whom I spent many days and nights working on the different courses, and to all those that, although not coursing the same degree, helped me become who I am today.

Lastly, I would like to thank Daniel Franco Puentes, my tutor and thesis director, for his support during this last 5 months.

REFERÈNCIES

- [1] Zaharia, Andra, “300+ Terrifying Cybercrime and Cybersecurity Statistics (2022 EDITION)”, Comparitech, September 21, 2022; <https://www.comparitech.com/vpn/cybersecurity-cyber-crime-statistics-facts-trends/>
- [2] Capers, Zach, “Zero Trust Improves Cybersecurity According to 99% of Companies That Adopt It”, Capterra, August 17th, 2022, <https://www.capterra.com/resources/zero-trust/>
- [3] Palo Alto Networks, “What is a Zero Trust Architecture”, Palo Alto Networks, October 29th, 2022, <https://www.paloaltonetworks.com/cyberpedia/what-is-a-zero-trust-architecture>
- [4] Microsoft, “Active Directory Domain Services Overview”, Microsoft, August 17th, 2022; <https://learn.microsoft.com/en-us/windows-server/identity/ad-ds/get-started/virtual-dc/active-directory-domain-services-overview>
- [5] Microsoft, “Microsoft by the numbers: adding up the stories that made Microsoft”, Microsoft, August 20th, 2022, <https://news.microsoft.com/bythenumbers/en/windowsdevices>
- [6] Palo Alto Networks, “What is Network Segmentation?”, Palot Alto Networks, October 20th, 2022, <https://www.paloaltonetworks.com/cyberpedia/what-is-network-segmentation>
- [7] Illumio, “Overview of Illumio Edge”, Illumio, October 29th, 2022, <https://docs.illumio.com/edge/22.11/Content/Guides/illumio-edge-usage/edge-overview/overview.htm>
- [8] ProjectManager.com, “Waterfall Model”, ProjectManager.com, October 29th, 2022, <https://www.projectmanager.com/guides/waterfall-methodology>
- [9] Atatus, “Agile Methodology”, Atatus, June 1st, 2021, <https://www.atatus.com/glossary/agile-methodology/>
- [10] Hoory Leeron, and Bottorf, Cassie, “Agile vs Waterfall: which project management methodology is best for you?”, Forbes, August 10th, 2022, <https://www.forbes.com/advisor/business/agile-vs-waterfall-methodology/>
- [11] Kaspersky, “Living off the Land (LotL) attacks”, Kaspersky, November 1st, 2022, <https://encyclopedia.kaspersky.com/glossary/lotl-living-off-the-land/>
- [12] Vue.js, “Themes, Creative Tim”, Vue.js, November 13th, 2022, <https://vuejs.org/ecosystem/themes.html>
- [13] Spring Initializr, Spring Initializr, November 20th, 2022, <https://start.spring.io/>
- [14] Protectera, “Zero Trust is an Approach not a Technology Product- History, Principles and Types”, Protectera, October 29th, 2022; <https://protectera.com.au/zero-trust-is-an-approach-not-a-technology-product/>
- [15] Illumio, “Illumio Core: Segmentation for on-premises and cloud data center workloads”, Illumio, December 30th, 2022, <https://www.illumio.com/products/illumio-core>
- [16] Microsoft, “What is PowerShell?”, Microsoft, December 30th, 2022, <https://learn.microsoft.com/en-us/powershell/scripting/overview?view=powershell-7.3>
- [17] Microsoft, “NetSecurity”, Microsoft, December 30th, 2022, <https://learn.microsoft.com/en-us/powershell/module/netsecurity/?view=winserver2012-ps>
- [18] Microsoft, “Windows Remote Management”, Microsoft, December 30th, 2022, <https://learn.microsoft.com/en-us/windows/win32/winrm/portal>
- [19] Microsoft, “Invoke-Command”, Microsoft, December 30th, 2022, <https://learn.microsoft.com/en-us/powershell/module/microsoft.powershell.core/invoke-command?view=powershell-7.3>
- [20] MYousufAli & Peter Geelen, “Active directory: GPO to Make a Domain User the Local Administrator for all PCs”, December 24th, 2022, Social Technet Microsoft, <https://social.technet.microsoft.com/wiki/contents/articles/7833.active-directory-gpo-to-make-a-domain-user-the-local-administrator-for-all-pcs.aspx>
- [21] Docker, “Package Software into Standardized Units for Development, Shipment and Deployment”, Docker, January 1st, 2023, <https://www.docker.com/resources/what-container/>
- [22] Crowdstrike, “WHAT IS ENDPOINT DETECTION AND RESPONSE (EDR)?”, Crowdstrike, January 1st, 2023, <https://www.crowdstrike.com/cybersecurity-101/endpoint-security/endpoint-detection-and-response-edr/>

- [23] IBM, “What is SIEM?”, IBM, January 1st, 2023,
<https://www.ibm.com/topics/siem>
- [24] Ericom, “What is Layered Security?”, Ericom, January 1st, 2023,
<https://www.ericom.com/whatis/layered-security/>
- [25] Fortinet, “What is DMZ network?”, Fortinet, January 1st, 2023,
<https://www.fortinet.com/resources/cyberglossary/what-is-dmz>
- [26] AWS, “Connect to the internet using an Internet Gateway”, AWS, January 6th, 2023,
https://docs.aws.amazon.com/vpc/latest/userguide/VPC_Internet_Gateway.html
- [27] AWS, “What is Elastic Load Balancing?”, AWS, January 6th, 2023,
<https://docs.aws.amazon.com/elasticloadbalancing/latest/userguide/what-is-load-balancing.html>
- [28] AWS, “What is Amazon EC2?”, AWS, January 6th, 2023,
<https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/concepts.html>
- [29] AWS, “What is Amazon Elastic File System?”, AWS, January 6th, 2023,
<https://docs.aws.amazon.com/efs/latest/ug/whatisefs.html>

APPENDIX

A.1 Gantt diagram

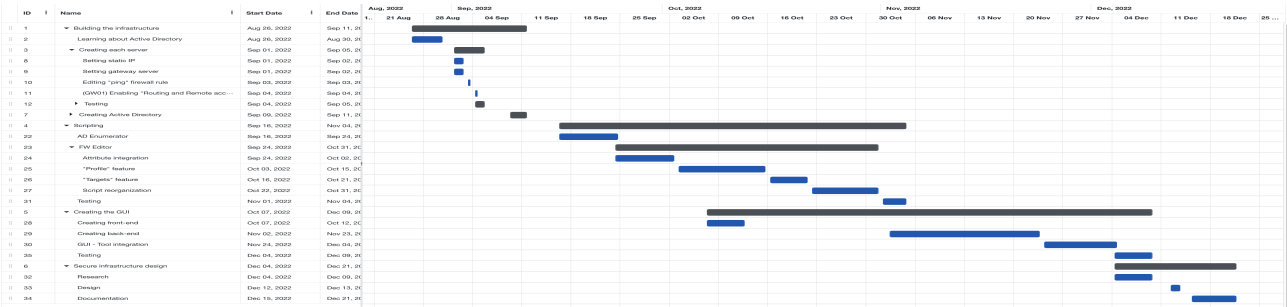


Fig. 15: Initial project's Gantt diagram