

Maria Lill Garvik, Synne Lindås & Fridtjof Bø  
Svendsen

# Exploring How To Secure MAC Authentication

Bachelor's thesis in Digital Infrastructure and Cybersecurity

Supervisor: Olav Skundberg

May 2023



Maria Lill Garvik, Synne Lindås & Fridtjof Bø  
Svendsen

# Exploring How To Secure MAC Authentication

Bachelor's thesis in Digital Infrastructure and Cybersecurity  
Supervisor: Olav Skundberg  
May 2023

Norwegian University of Science and Technology  
Faculty of Information Technology and Electrical Engineering  
Department of Computer Science





# Exploring How To Secure MAC Authentication

Fridtjof Bø Svendsen, Maria Lill Garvik, Synne Lindås

19/05/2023

# Abstract

The evolution of network architecture is changing rapidly with the growing use of advanced network management platforms from large network companies. However, the nature of network implementation in an organization still necessitates consideration for older devices when implementing new architecture. Due to these changes, the current solutions for handling these devices are not yet fully secure. Lack of secure authentication of legacy devices therefore becomes a security issue.

This thesis will examine how one can improve the security of authenticating devices like legacy devices and IoT on a wired network. This was accomplished by studying existing literature and solutions, as well as performing some tests.

The results of this project show that some effort can be made to improve the security of the authentication process. The solution includes adding extra attributes in the authentication process to help determine if the device wanting access is the one it claims to be. However, it still remains difficult to completely secure authentication of IoT and older devices on the network.

# Sammendrag

Utvilkingen av nettverksarkitektur endrer seg hurtig gjennom økende bruk av avanserte plattformer for nettverksadministrasjon fra store nettverksselskaper. Det må likevel fremdeles tas hensyn til eldre enheter i nett når man implementerer ny arkitektur. Disse endringene mangler komplette løsninger for å håndtere disse enhetene. Manglende evne til å autentisere eldre enheter vil derfor skape utfordringer innen sikkerhet.

Denne oppgaven vil se på hvordan man kan forbedre sikkerhet knyttet til autentisering av eldre enheter og IoT på et kablet nettverk. Dette ble oppnådd ved litteraturstudie og testing.

Resultatene fra dette prosjektet viser at noen løsninger kan implementeres for å forbedre sikkerheten rundt autentiseringsprosessen. Løsningen innebærer å legge til ekstra attributter i autentiseringsprosessen som kan hjelpe med å bekrefte om enheten som vil ha tilgang faktisk er den enheten den sier den er. Det forblir derimot vanskelig å komplett sikre autentisering av IoT eldre enheter på et nettverk.

# Preface

This bachelor thesis was written as the final assignment of our bachelor's degree in Digital Infrastructure and Cybersecurity at the Norwegian University of Science and Technology (NTNU). The thesis explores MAC-Authentication in an enterprise network and how it can be implemented in a secure manner. We were motivated to write this thesis as secure access to networks is of utmost importance, and most organizations will have legacy devices and other devices not supporting the safest means of network access technologies. Our hope is that this thesis will be of help to NTNU and other organizations in increasing the knowledge and understanding regarding secure authentication of devices.

We want to thank our supervisors at NTNU IT Knut Carlsen, Håvard Ose Nordstrand and Øyvind Smith-Øvland for answering all our questions, taking your time explaining concepts, and giving good advice on the thesis. We also want to thank our wonderful supervisor at NTNU, Olav Skundberg, for the excellent follow-up and all the feedback throughout the process. At last, we want to thank all our friends and family who took the time to read our thesis. We could not have finished the thesis without you.



# Contents

<b>Abstract</b> . . . . .	<b>I</b>
<b>Sammendrag</b> . . . . .	<b>II</b>
<b>Preface</b> . . . . .	<b>III</b>
<b>Contents</b> . . . . .	<b>IV</b>
<b>Figures</b> . . . . .	<b>VI</b>
<b>Acronyms</b> . . . . .	<b>VII</b>
<b>Glossary</b> . . . . .	<b>IX</b>
<b>1 Introduction</b> . . . . .	<b>1</b>
1.1 Background and Motivation . . . . .	1
1.2 Thesis Topic . . . . .	2
1.2.1 Research questions . . . . .	2
1.3 The Partner Organization and Case . . . . .	2
1.4 Scope . . . . .	3
1.5 Thesis Layout . . . . .	4
<b>2 Theory</b> . . . . .	<b>5</b>
2.1 Access Control . . . . .	5
2.2 Zero Trust Network Architecture . . . . .	6
2.3 MAC Authentication . . . . .	6
2.3.1 MAC Address . . . . .	6
2.3.2 MAC Authentication Bypass . . . . .	7
2.4 Vulnerabilities in MAC Authentication . . . . .	9
2.4.1 MAC Spoofing . . . . .	9
2.4.2 Lack of User Identification and Accountability . . . . .	9
2.4.3 Vulnerabilities in RADIUS . . . . .	10
2.5 Other Authentication Alternatives and Technologies . . . . .	10
2.5.1 802.1x . . . . .	10
2.5.2 Port Security . . . . .	12
2.5.3 Additional Security Measures . . . . .	12
2.6 Software Defined Networking and Access . . . . .	14
2.6.1 Secure Group Tagging . . . . .	14
2.6.2 Segmentation . . . . .	15
2.6.3 Campus Fabric Architecture . . . . .	15
2.7 Cisco ISE . . . . .	16
2.7.1 Policies in ISE . . . . .	16
2.7.2 Using MAB with RADIUS and Cisco ISE . . . . .	17
2.7.3 Cisco ISE APIs . . . . .	18
<b>3 Method</b> . . . . .	<b>19</b>
3.1 Chapter Outline . . . . .	19

3.2	Literature Study - Information Gathering and Research . . . . .	20
3.2.1	Search Engine . . . . .	20
3.2.2	Criteria for Selecting Sources . . . . .	20
3.3	Testing . . . . .	21
3.3.1	Sandboxing . . . . .	21
3.3.2	Scripting . . . . .	23
3.3.3	Test Case One - Access Attempt Without Custom Attributes . . . . .	24
3.3.4	Test Case Two - Access Attempting With Custom Attributes . . . . .	26
3.4	Criteria for Security Evaluation . . . . .	27
<b>4</b>	<b>Results . . . . .</b>	<b>30</b>
4.1	Testing Prerequisites . . . . .	30
4.1.1	Sandboxing . . . . .	30
4.1.2	Scripting . . . . .	30
4.2	Case One: Access Attempt Without Custom Attributes . . . . .	37
4.3	Case Two: Access Attempting With Custom Attributes . . . . .	39
4.3.1	Location Attribute . . . . .	39
4.3.2	Hardware Attribute . . . . .	41
<b>5</b>	<b>Discussion . . . . .</b>	<b>42</b>
5.1	Literature Provider and Year of Publication . . . . .	42
5.2	The Script . . . . .	42
5.3	Case One - Access Without Custom Attributes . . . . .	43
5.4	Case Two - Access With Custom Attributes . . . . .	43
5.4.1	Location Attribute . . . . .	44
5.4.2	Hardware Attribute . . . . .	44
5.5	Other Methods for Securing MAB . . . . .	45
5.5.1	Port Security . . . . .	45
5.5.2	Physical Security . . . . .	45
5.5.3	Network Segmentation and Monitoring . . . . .	46
<b>6</b>	<b>Conclusion . . . . .</b>	<b>47</b>
6.1	Summary Conclusion in Relation to the Research Questions . . . . .	47
6.1.1	Research Question One . . . . .	47
6.1.2	Research Question Two . . . . .	47
6.1.3	Research Question Three . . . . .	48
6.2	Thesis Summary . . . . .	48
6.3	Future Work . . . . .	49
	<b>Bibliography . . . . .</b>	<b>50</b>
<b>A</b>	<b>Python Code . . . . .</b>	<b>54</b>
A.1	Script: Main Interface . . . . .	55
A.2	Script: Add Endpoint . . . . .	56
A.3	Script: Edit and Delete Endpoint . . . . .	57
A.4	Script: Add Endpoints in Bulk . . . . .	59
A.5	Script: MAC-address Validator . . . . .	60
A.6	Example JSON file . . . . .	61

# Figures

1.1	General Network Hierarchy . . . . .	3
2.1	MAB Overview . . . . .	8
2.2	RADIUS Package . . . . .	8
2.3	802.1x Communication . . . . .	11
2.4	VLAN Example . . . . .	13
2.5	Network Overlay . . . . .	15
2.6	Campus Fabric Overview . . . . .	16
2.7	Overview of Cisco ISE . . . . .	17
3.1	Method Flow . . . . .	19
3.2	Sandbox Topology Overview . . . . .	22
3.3	Example of VNs In Sandbox . . . . .	22
3.4	MAB Policy Set . . . . .	25
3.5	Policy Condition: Check Endpoint Group . . . . .	25
3.6	Identity Group Condition . . . . .	25
3.7	Location Condition . . . . .	26
3.8	Risk Environment Overview . . . . .	28
4.1	Script Main Interface . . . . .	31
4.2	List All Endpoints in a Group with Script . . . . .	32
4.3	List Information of Specific Endpoint with Script . . . . .	32
4.4	Endpoint Added With Script . . . . .	33
4.5	Endpoints Added to ISE . . . . .	33
4.6	Network Access Before and After Registration . . . . .	34
4.7	Name Field Overwritten . . . . .	34
4.8	Endpoints Added in Bulk . . . . .	35
4.9	Edit Endpoint . . . . .	35
4.10	Delete Endpoints . . . . .	36
4.11	Custom Attribute . . . . .	36
4.12	Original MAC Address Device Used for Spoofing . . . . .	37
4.13	MAC address After Spoofing . . . . .	37
4.14	Successful Spoofing . . . . .	38
4.15	Shows the Location Attribute of Registered Device . . . . .	39
4.16	Spoofing Failed With Location Attribute . . . . .	40
4.17	Hardware Attribute Based on MAC Address . . . . .	41
A.1	Bulk add JSON file template . . . . .	61

# Acronyms

- AAA** Authentication, Authorization, and Accounting. 9, 16
- ABAC** Attribute Based Access Control. 5
- AC** Access Control. 5, 6
- ACI** Application Centric Infrastructure. 23
- API** Application Programming Interface. 4, 18, 23, 30, 43
- BYOD** Bring Your Own Device. 6
- CRUD** Create, Read, Update, Delete. 18
- CSV** Comma-separated Values. 23
- DAC** Discretionary Access Control. 5
- EAPoL** Extensible Authentication Protocol over LAN. 11
- ERS** External Restful Services. 18, 30
- GUI** Graphical User Interface. 23
- HTML** HyperText Markup Language. IX
- IoT** Internet of Things. 1, 2, 9, 11, 49, I, II
- ISE** Identity Service Engine. 4, 14, 16–18, 21, 23–27, 30, 32–37, 39, 41–44, 48, 49
- JSON** JavaScript Object Notation. 23, 30, IX
- LISP** Locator/Identifier Separation Protocol. 15
- MAB** MAC Address Bypass. 7, 9, 10, 12, 14, 17, 18, 21, 24, 26–29, 36, 37, 42–46, 48, 49
- MAC** Media Access Control. 2–4, 6, 7, 9, 11, 12, 19–21, 23–28, 32–34, 37–41, 43–49
- OS** Operating System. 16
- OSI** Open System Interconnection. 6, 7

**OUI** Organizationally Unique Identifier. 6

**RADIUS** Remote Authentication Dial-In User Service. 7, 9–11, 17, 18, 42

**RBAC** Role Based Access Control. 5

**RLOC** Routing Locator. 15

**SDA** Software Defined Access. 14, 15, 21

**SDN** Software Defined Networking. 14

**SGT** Secure Group Tagging. 14–18

**UX** User Experience. 23

**VLAN** Virtual Local Area Network. 11, 12, 14

**VM** Virtual Machine. 23

**VN** Virtual Network. 15, 17, 21, 24, 45, 46

**VXLAN** Virtual Extensible LAN. 16

# Glossary

**802.1x** An authentication method for wired and wireless network access where the connecting device must authenticate the switch or access point they connect to with credentials or a certificate (Jha [1]). . 3

**ACL** Access Control List is a list of rules applied to switches and routers allowing or denying access to a network.. 14

**CIA Triad** CIA, or Confidentiality, Integrity and Availability, is an important guiding principle for information security. The goal is to guarantee protection of stored information by keeping the data from unauthorized access (confidentiality), keeping the data unaltered (Integrity) and keeping the data accessible for use (Availability).. 2

**Control Plane** The part of the network controlling forwarding of packets, and includes creation of routing tables Cloudfare [2] . 15

**Data Plane** Also called the Forwarding Plane, is the part of the network that forwards packets Cisco [3] . 15

**Least Privilege** A security concept where a user in a system is given the absolute minimum levels of access needed to perform the necessary functions. 5, 12

**MAC Address Bypass** An authentication method where network access is determined based on a device' MAC address (Cisco [4]). 11

**MAC spoofing** The act of altering the MAC address of a device. 9

**Policy Plane** Part of the network used for security and segmentation Cisco [5]. . 15

**REST** Representational State Transfer is a style of API formatting that contains a number of restraints which increases uniformity where implemented. A REST API, also known as a RESTful API, delivers information via HTTP in a format like JSON, HTML or plain text . 18, 23

**VLAN** Virtual Local Area Networks is a way to logically divide one or more local area networks. 12, 14

**VRF** Virtual Routing and Forwarding (VRF) can be explained as Layer 3 VLANs. Multiple virtual routers may exist on one router, with VRFs on one or multiple interfaces. Packets within a VRF are only forwarded between interfaces with the same VRF [6] . 15

# Chapter 1

## Introduction

### 1.1 Background and Motivation

In today's society, everything is connected, and a reliable network is essential for organizations to produce value. Due to society's heavy reliance on network communication, it becomes an attractive target for threat actors. A particular vulnerable part of the network is the access layer. This is accessible for anyone located in an organization's public cafeteria or waiting room, and does not require strict access control like the data center.

It is possible for attackers to gain access to networks via spoofing attacks. Compared to ransomware attacks where the threat actor might want the attacked organization to know that an attack is happening, a spoofing attack might go unnoticed for a long time. It is estimated that an attacker can hide in the network for an average of 146 days before the breach is detected [7]. There are many technologies to ensure access to authorized users and devices only, and that limits the chance of spoofing attacks. However, the technologies available vary depending on the age and design of the network architecture and connected devices.

Today's network consists of many types of devices, some of these are legacy and other devices that don't support modern authentication and security methods. Threat actors evolve and find new ways to gain access to networks and devices that can't keep up with this development. As a result, they can be considered a security vulnerability in the network since they can be used as an entry point for the threat actor. To ensure the legitimacy of a device, authentication can be done using certificates. However, to be able to authenticate using certificates, the device needs to have the resources and processing power to process these types of authentication and requests, as well as support the protocols being used. Therefore, the devices that do not support secure authentication methods present a problem for the security in an organization and society in general.

Some relevant examples of devices with the problems and vulnerabilities mentioned are legacy devices, Internet of Things (IoT) and equipment like multimedia devices. Multimedia devices are common in organizations with a similar nature to universities (e.g. NTNU), using this equipment for streaming of lectures and the like. The use of IoT devices are also on the rise. According to a report by IEEE, IoT devices are particularly vulnerable to malicious intentions and it's therefore important to focus on the security aspects of these devices [8].

The main focus of this thesis will be look into how one can prevent unauthorized access to the network, thus ensuring confidentiality on the network. Availability is another concept of importance as a network has to be available to legitimate users and devices. These concepts corresponds with the CIA Triad. When the security measures are implemented, authorized access to the network must still be maintained to the highest possible degree.

It is common for organizations to have legacy devices in their network, and the increased usage of IoT devices together with the mentioned security issues are motivators for this bachelors project. This is a topic affecting many parts of society and organizations, making this topic very relevant. Especially in the stage of replacing legacy devices and updating the network to contain new and more IoT devices.

A much used authentication method for legacy and IoT devices is MAC Authentication, but this authentication method is considered insecure. More secure network access control methods exist, but most organization will have devices that only support MAC Authentication as devices like the ones mentioned above does not support more secure authentication methods. With such a widespread problem, how can organizations safely use MAC authentication as a mean of access control?

## **1.2 Thesis Topic**

The thesis topic for this project is to explore secure access to wired networks. The focus will be on MAC authentication and how to use it in a secure manner. Three research questions have also been developed to help answer the thesis' topic.

### **1.2.1 Research questions**

#### **Research question One**

To what extent can MAC authentication be used to prevent unauthorized access to wired networks?

#### **Research question Two**

Are there solutions combined with MAC authentication that could make MAC authentication more secure?

#### **Research question Three**

What other alternatives exists for MAC authentication?

## **1.3 The Partner Organization and Case**

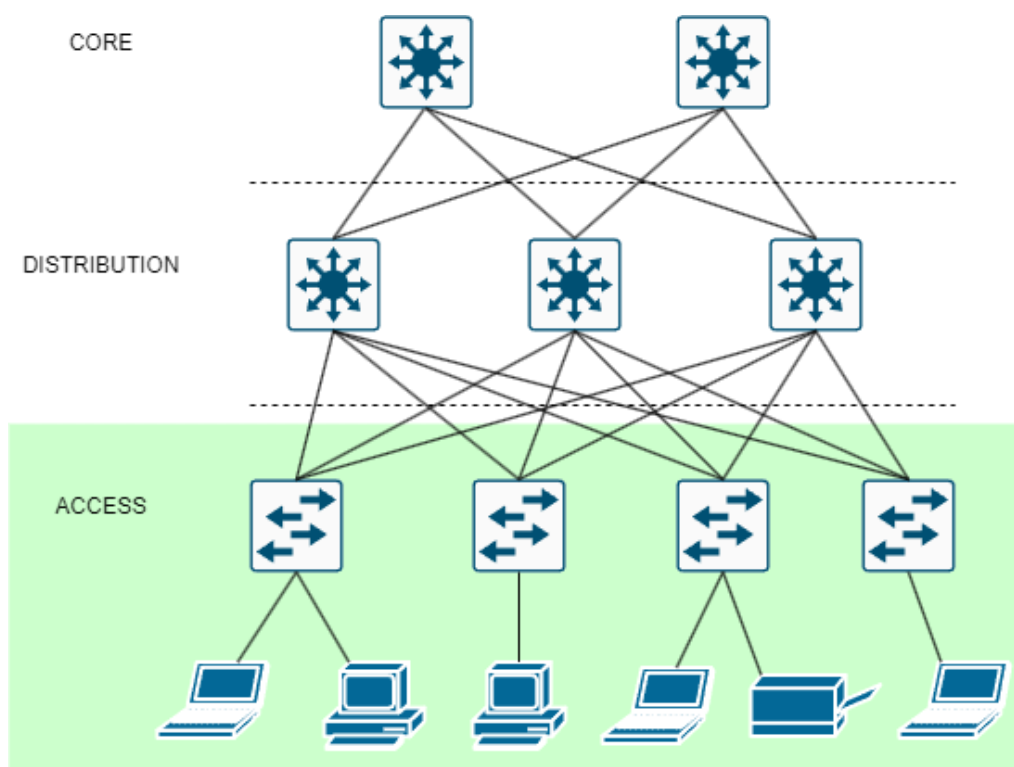
This bachelor project was accomplished in cooperation with NTNU-IT. They deliver services to almost 50 000 students and employees at NTNU, and is in the process of upgrading their enterprise network infrastructure. This includes introducing new concepts for organizing the network and improving their security. Further explanations of the new network architecture will be in Chapter 2.6.



As a part of the upgrade, a new way of registering devices that are to use MAC authentication on the network was needed. Currently these MAC devices are manually registered in Active Directory, but a new, more efficient and user friendly method was wanted. A part of this thesis was therefore to contribute with a prototype in the form of a script for managing MAC address registration of devices that are to be given access on the new network. Another goal was to share the thoughts and solutions on how to make MAC Authentication as secure as possible. NTNU-IT provided resources such as a sandbox environment for testing, technical knowledge and organizational support.

## 1.4 Scope

For this thesis the focus will be on the access network, and the scope is limited to wired access and non 802.1x compatible devices. This excludes any wireless technologies and devices that support other authentication methods than MAC authentication. The focus areas are MAC authentication and the security issues and possibilities this authentication method has, which includes the use of additional attributes together with the MAC address for authentication. The thesis only focuses on two of these attributes; location and hardware. The network technologies, devices and solutions used throughout the project are all from Cisco, as they are the main supplier of network equipment and solutions to NTNU. All testing and work was done at the access layer of the network, depicted in figure 1.1 below.



**Figure 1.1:** A general network topology. This thesis will focus on the access layer of the network, marked with green.

## 1.5 Thesis Layout

### **Chapter 1: Introduction**

Chapter one provides the background of the project, the thesis topic and its research questions, as well as an introduction to the Partner Organization and scope of the project.

### **Chapter 2: Theory**

In chapter two the theory behind secure network access and the background for later methods and analysis are discussed. This includes MAC authentication and 802.1x, as well as important security principles and best practices in network security that are presented to support the thesis' security discussion. Some aspects of Cisco Software Defined Access will also be explained as this is the basis of NTNU's new network. Cisco ISE and policies in ISE will also be explored, and lastly APIs available through Cisco are presented.

### **Chapter 3: Method**

Chapter three will define the method used for answering the research questions. This includes the scientific methodology for research and the work involved in activities like scripting, testing and analysis of the result. The chapter will also describe the criteria for evaluating the tested methods.

### **Chapter 4: Results**

Chapter four presents the results from the tests described in chapter three. This also includes the script and its use, and how it works with the current authentication systems for NTNU-IT.

### **Chapter 5: Discussion**

Chapter five discusses the results gathered from the research in chapter two (Theory) and the results from the testing in chapter four (Results).

### **Chapter 6: Conclusion**

In chapter six the findings of the thesis, answers to the thesis topic and research questions will be summarized, as well as future work.

# Chapter 2

## Theory

This chapter introduces the relevant theory and concepts for the thesis, which will be used as a foundation for the thesis' testing and discussions.

### 2.1 Access Control

The basis of Access Control (AC) in network security is proper identification, authentication and authorization. This means that every device that requests access to the network must be identified, and this identity must be authenticated. After the authentication, the proper levels of access to the network must be authorized. With these mechanics, an organization is able to regulate who or what is permitted access to resources within a system. The most common method for configuring AC is setting policies with defined roles which allow proper levels of access necessary for each role. However, this varies based on the type of AC used.

Access control can be set up in different ways based on an organization's needs and security concerns [9].

- Discretionary Access Control (DAC), is AC based on user discretion where the owner of a resource is responsible for managing access. This method is not suited for larger structures or where scalability is an issue.
- Role Based Access Control (RBAC), is AC based on defined roles where the organization have determined which resources should be available for each role. This is the most widely used variant of AC due to the high levels of control it gives a system administrator regarding automation, scalability and security.
- Attribute Based Access Control (ABAC), is AC based on several variables regarding the user, device or environment. Examples of attributes can be role, user or device location or the time of day. Compared to RBAC, ABAC gives a higher degree of flexibility in terms of configuring necessary permissions due to the way multiple different types of attributes give more control than RBAC.

The concept of Least Privilege is important in access control for minimizing risk regarding unauthorized access to a network. In the case of a threat actor achieving some level of access, the damage can be controlled by ensuring the access available to the intruder is as minimal as possible. An evolution of this concept is zero trust network architecture [10].

## 2.2 Zero Trust Network Architecture

Zero trust is a paradigm shift from static network defense to focus on users and resources [11]. This involves implementing policies based on no implicit trust between elements in the network, like users. Previously, a user could expect inherent trust based on factors like physical or network location. Still, with zero trust access has to be verified to a much higher degree than with traditional architecture. AC functions like authentication and authorization are performed by software before a user session. The main advantage of zero trust is increased security through the prevention of lateral movements, where the threat actor moves through the network. This is more beneficial than traditional perimeter-based defense, where an attacker could easier initiate lateral movements or privilege escalation once inside the "secure" network area. Zero trust principles summarized as [12]:

1. All data sources and computing services are treated as organizational resources, which must be secured.
2. Despite the network location of the access request, all communication is secured. No trust is granted automatically, and no assets requesting access are trusted by default.
3. Access to resources is granted per session only.
4. Access is determined based on device characteristics, and behavioral and environmental attributes.
5. Least privilege applies.
6. Access is not granted statically but continuously re-evaluated.
7. Information about assets, network infrastructure, and communications is collected and used to improve security.

The principles of zero trust utilizes attribute-based access control, but with the concept of no resource being inherently trusted. This allows a more secure implementation of external elements often used in a modern work environment like Bring Your Own Device (BYOD). But to control which users and devices can access what resources, a method for authenticating devices and users is needed. One of them being MAC authentication.

## 2.3 MAC Authentication

### 2.3.1 MAC Address

Media Access Control (MAC) is a type of address given to devices to help identify them, and is unique to the device. The address consists of 12 hexadecimal characters (or 48 bits) with different conventions for how to divide the address. A common convention is separating the address in groups of two using dash (-) or colon (:). Two less common conventions are separating the address in groups of four with a dot (.) and not separating the address. An example of a MAC address separated by a dash is 03-C2-D0-53-C3-35 [13]. The MAC consists of two parts; the first 24 bits is the Organizationally Unique Identifier (OUI) representing the device vendor, and the last 24 is a unique number for identifying the device. The OUI is assigned to vendors by the IEEE [14].

The MAC address is used for device identification and therefore works in the data link layer (second layer) in the Open System Interconnection (OSI) network model. The Data Link Layer is responsible for the data that is transferred between nodes over the physical layer (layer one

of the OSI model) [15]. The MAC addresses are also used for locating devices on the internet, troubleshooting and to ensure end-to-end communication [16].

Since the MAC address is uniquely assigned to the device, it can be used to identify the device and therefore also in authentication processes. How can this be done and is it considered a reliable way of authenticating?

### 2.3.2 MAC Authentication Bypass

MAC Address Bypass (MAB) is a method for authentication where connecting devices are given or denied access based on their MAC address [17]. MAB enables and disables ports dynamically based on the MAC address of the connected device, where the port is disabled for unauthenticated and enabled for authenticated devices. Before a device is authenticated, all traffic are blocked except from the first packet sent, from which the switch learns the MAC address, and then uses this to authenticate with MAB [4]. After the authentication the switch will filter the traffic from the port based on the authenticated MAC address.

#### Authentication Using MAB

MAB authenticates the device using a centralized repository for MAC addresses, access rights and policies, using the RADIUS protocol. The switch authenticates the device by sending an RADIUS Access-Request message with the MAC address. If the device is verified, a RADIUS Access-Accept message is returned, and if not a RADIUS Access-Reject is returned. The Access-Request message contains different attributes, including a username, password and a calling-station-id. The calling-station-id tells the authenticator where the request came from [18]. Different types of RADIUS servers will use different attributes to verify the device [4]. By default, the username and password attributes contain the MAC address of the device [19].

The last RADIUS response is RADIUS Access-Challenge. This message is sent by the RADIUS server to the switch to request more information about the device to be able to authenticate it. The switch will then send a new RADIUS Access-Request message to the server with the requested information [20]. Figure 2.1 and 2.2 below shows the communication process during an authentication and the different fields in a RADIUS packet.

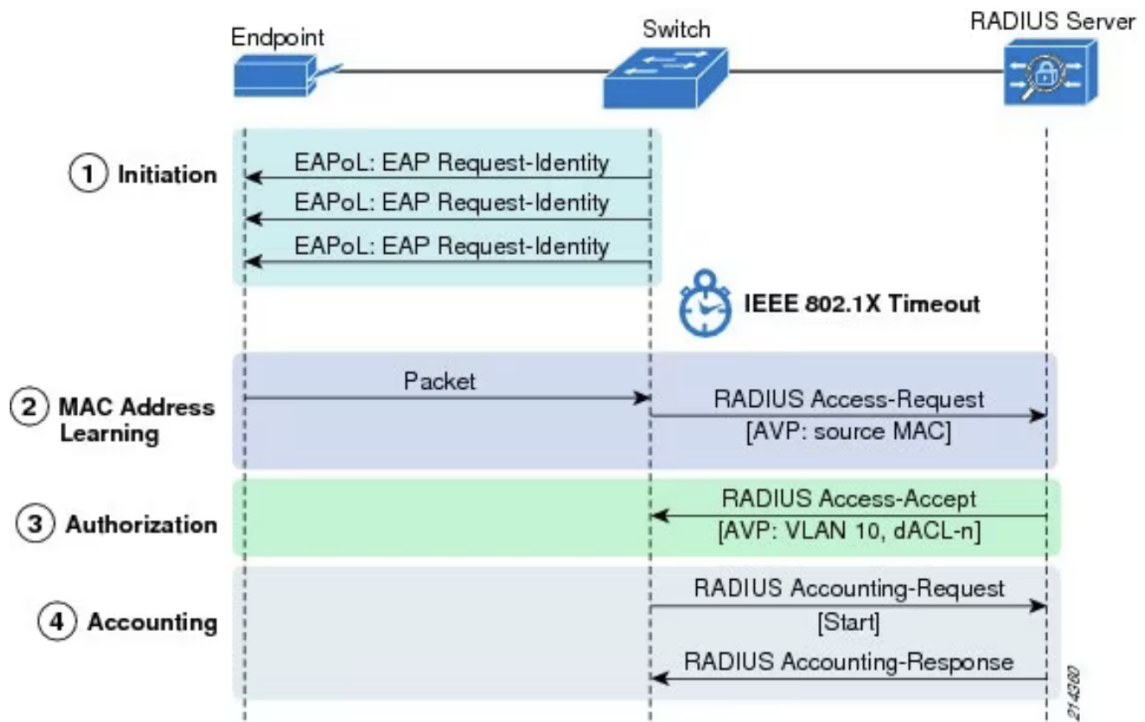


Figure 2.1: The figure taken from Cisco’s MAB deployment guide [4] and shows an overview of the MAB authentication process, starting at point two (2) with "MAC Address Learning".

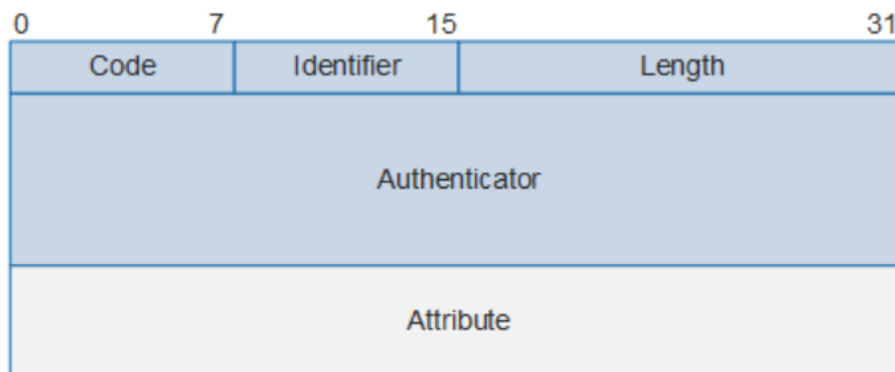


Figure 2.2: The figure shows a RADIUS packet with the possible fields [21].

### Modes in MAB

MAB has multiple modes for controlling the access of devices on a port. These modes are; Single-host, Multi domain authentication host, Multi-host and Multi-authentication host mode. Single-host only authenticates and gives access to one source MAC address. In this mode, only the verified MAC address can connect. If more tries to connect, this will cause a security alert that shuts down the affected port. Multi-domain authentication host mode works the same as single-host mode, but allows two source MAC addresses on the port. On the other hand, Multi-authentication host mode allows multiple MAC addresses, but each one has to be authenticated separately with MAB. Similar the Multi-host mode allows multiple addresses, but only the first has to be authenticated, and the rest will automatically be permitted on the network [22].

### RADIUS Server

MAB also makes use of a RADIUS Server and RADIUS protocols to authenticate the devices and assign the appropriate permissions. RADIUS server is a type of AAA server, i.e. a server that is responsible for Authentication, Authorization and Accounting [17]. The RADIUS Server receives the authentication information from the device and verifies this with its records, and thus functions as both a centralized device and Authentication, Authorization, and Accounting (AAA) manager [23].

### MAB and RADIUS Inactivity Timer

Inactivity timer is a function that can be used by either MAB or RADIUS, and lets the administrator configure a time period before the session of an authenticated device expires, i.e. the device has to be re-authenticated and a new session has to be started. This is a security function preventing a port remaining open or someone gaining access through an open port linked to an inactive device. The timer is statically assigned with the MAB function, or the RADIUS package field called Idle-Timeout attribute can be used to dynamically assign the timer. Using RADIUS gives more control and flexibility over for example the timer length and the device classes to assign the timers [4].

## 2.4 Vulnerabilities in MAC Authentication

MAB is a much used authentication method for devices like legacy, multimedia and IoT devices. However, MAB has multiple vulnerabilities. These include spoofing of the MAC address, lack of user identification, and lack of accountability, as well as vulnerabilities in the RADIUS protocol.

### 2.4.1 MAC Spoofing

MAC spoofing is the act of altering the MAC address of a device. Hackers may do this to gain access to a network by altering their MAC to match an authorized device on the network, and it is not a complicated procedure [14]. With Linux one can easily install *Macchanger* and with the following command change the MAC address to one's liking [24]:

```
sudo macchanger -m custom-address interface
```

### 2.4.2 Lack of User Identification and Accountability

MAB only authenticates devices based on their MAC addresses and does not provide any mechanism for user identification. This means that once a device gains network access, anyone who

has physical access to that device can potentially use it to access the network without requiring any additional authentication. This lack of user identification makes it difficult to trace the source of any potential security breaches.

Since MAB does not provide any user identification or authentication, it is difficult to hold users accountable for their actions on the network. This can make it challenging to investigate security incidents or track down the source of any malicious activity.

### 2.4.3 Vulnerabilities in RADIUS

Not only has MAB vulnerabilities, the RADIUS protocol itself has multiple vulnerabilities. Two examples of where these vulnerabilities can be found are a user-password vulnerability and a shared secret vulnerability. Since RADIUS is central to the use of MAB, vulnerabilities regarding one also affect the other.

RADIUS uses a shared secret called the Request Authenticator, which have to be a unique and non-predictable to be used as a security measure, but many implementations uses insufficient random number generators, leading to a poorer level of protection [25].

The Request Authenticator and a MD5 hashing algorithm is used by RADIUS to encrypt sensitive attributes like User-Password, but his is the only attribute encrypted everything else in the package is in plain text. The encryption used for the password, MD5, is used as a stream cipher primitive, which it is not designed for and might therefore lead to a flawed system [25].

## 2.5 Other Authentication Alternatives and Technologies

Other solutions than MAB as an authentication method exists, where some are considered more secure. The solutions discussed in this chapter is the 802.1x standard and Port Security, in addition to security considerations that should be assessed as additional security when using MAB.

### 2.5.1 802.1x

802.1x is an IEEE standard for Layer two Access Control, both wired and wireless, where authentication is based on the identity of the user or device wanting to connect to the network. To use 802.1x, three components are needed:

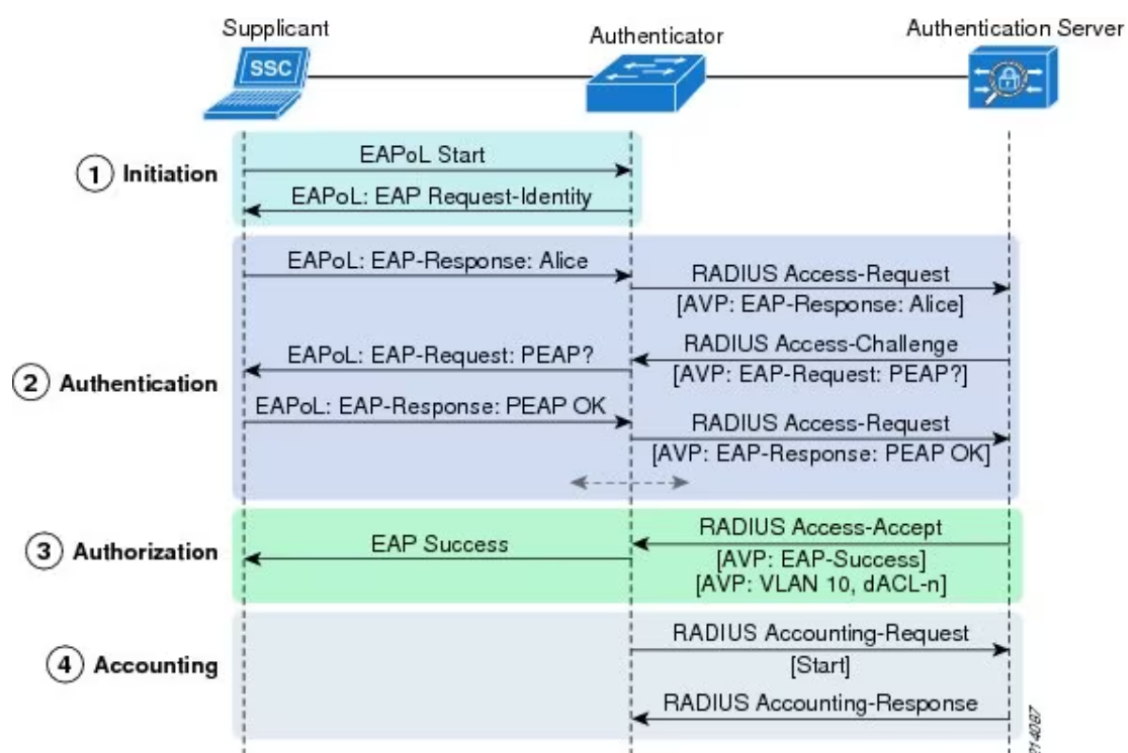
- *Supplicant/client*: The supplicant submits the credentials needed for authentication and runs as a client on the endpoint.
- *Authenticator*: The network device, for example an access switch, that relays the supplicant's credential to the authentication server.
- *Authentication Server*: A server that validates the received credentials which determines the user's or device's level of network access. This is most commonly a RADIUS server.

Together with the three components, a backend identity database is almost always used. An example is Microsoft Active Directory, as this already has identities and additional attributes stored, such as Security Groups memberships. This relieves the authentication server of



managing credentials, and the additional attributes can be used to make decisions about permissions the devices shall have on the network [26] [27].

On wired networks, the authentication process starts when the access switch detects a change on the 802.1x enabled port. The authenticator then requests credentials from the supplicant, which sends a response packet in return. The response package is forwarded from the authenticator to the authentication server with the RADIUS protocol, and the authenticator is then notified if the supplicant shall have access or not. The authenticator informs the supplicant of the status, and places it into an authorized VLAN should the authentication process be a success [28].



**Figure 2.3:** Figure depicting communication between components during the 802.1x authentication process. Figure from Cisco's Wired 802.1X Deployment Guide [26]

Should the connecting endpoint not support 802.1x, the port will try to authenticate with 802.1x first anyway. When the endpoint fails to respond to the Extensible Authentication Protocol over LAN (EAPoL) packet, can be configured to work as a failover (see 2.3.2). If the RADIUS Server approves the MAC address, the endpoint is considered authenticated on the port and is now connected [29].

What makes 802.1x safer than authentication based only on the MAC address of the device is the need for a password or certificate, which are not as easily stolen or faked compared to spoofing a MAC address. However, 802.1x requires client-side support, meaning the authentication method mostly works on operating systems like Windows, iOS, Linux etc., but has often less coverage on devices running embedded and proprietary operating systems like printers, IoT and the like [30].

### 2.5.2 Port Security

Port Security is a feature on switches where all traffic into a port can be blocked should it not originate from a MAC address specified on the port, or a given amount of MAC addresses allowed on the port is exceeded. If a violation occurs, the port can be configured to do one of the following two actions; Shutdown where the port shuts down for a specified time or permanently, or Restrict where packets from the insecure host are dropped [31]. This solution is also susceptible to MAC spoofing. However, should someone try connecting to a port and fail, it stops them from trying a second time, now maybe with a spoofed address, as the port may be shut down until manually put up again.

### 2.5.3 Additional Security Measures

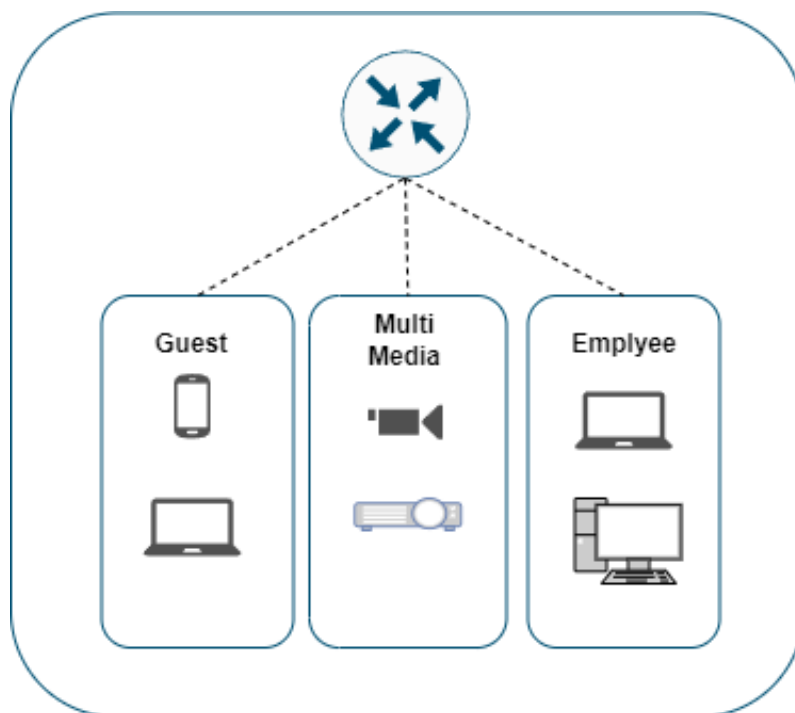
Should there be no better alternative to MAB, one should be aware of the possible consequence and implement necessary security measures accordingly. This includes being able to detect and limit the impact of an incident.

#### Network Monitoring

Network monitoring is vital to maintain network integrity and ensuring that the network functions as intended. By monitoring the network, one can get early warnings on traffic and device patterns [32], which in case of this thesis, might help detect a spoofing attack. If a printer suddenly starts contacting resources or domains it has never contacted before and has no need to communicate with, it might be a sign of someone having spoofed the printer. This is called anomaly detection, and can be done by solutions like Intrusion Detection Systems. Normal behavior and patterns are saved in profiles, and current activities are checked against the normal profiles [33]. The detected anomalies can either be treated passively or actively. When done passively, the system logs the anomaly and sends an alert. If it reacts actively, the system can reset connections or reprogram firewalls to block the malicious traffic.

#### Network Segmentation

Should an incident occur, it is important to limit the impact as much as possible. One way to do this is by ensuring that even though the threat actor gained access to the network, it has not gained access to the whole network but only a small part. This is done with segmentation. The network segments, are often made using VLANs, where resources with similar security requirements are placed in the same VLAN (see figure 2.4) [34]. Segmentation also helps implement the Principle of Least Privilege, as VLANs should also be made based on what resources a given device or application group needs. If a threat actor gains access to one segment and it's resources, they won't necessarily have gained access to anything critical.



**Figure 2.4:** The figure is depicting a logical overview of a segmented network, where different devices and users are placed in different VLANs based on security and resource requirements.

## 2.6 Software Defined Networking and Access

This section explains the concepts behind Software Defined Networking and Software Defined Access, as these concepts are the backbone of NTNU's new network.

In traditional networking a network device contains of two planes: The Control Plane and the Data Plane. The Control Plane works as the brain of the device, making forwarding decisions. The data plane uses the information from the control plane to forward traffic. In Software Defined Networking (SDN) the Control Plane is moved to a centralized location, meaning forwarding decisions and device management is moved, leaving more CPU capacity to the device and easing network management for administrators.

Software Defined Access (SDA) uses the principle of SDN and places the access network under management of a centralized network controller. The primary goal of SDA is to improve network security by being able to push fine-grained access control to the access network, in addition to ease of management with centralized control [35].

Segmentation of networks and proper access control to the segments are crucial parts of managing networks, but as a network grows or merges with other networks, the traditional network will encounter some problems. Traditional network segmentation are based on placing endpoints in VLANs and using ACLs to control access. Should the organization merge in any way where new VLANs, IP-ranges and ACLs are needed, this can lead to ACL exhaustion on devices, in addition to the necessary administrative work. With Cisco SDA segmentation is done with Secure Group Tagging (SGT).

### 2.6.1 Secure Group Tagging

Instead of using the locations and IPs of users and devices to place them into VLANs for access control, Cisco ISE uses Secure Group Tagging (SGT), also called Scalable Group Tagging. When a device gets authenticated by ISE through 802.1x or MAB when connecting to a network, it is assigned a SGT [36]. The SGT is a 16-bit value that are that is used in the network in two ways:

- *Inlined*: The SGT is included in the data packet. When the next-hop device receives data packet, it forwards it or enforces policies based on the SGT.
- *SGT Exchange Protocol (SXP)*: When network devices does not support SGT, the TCP-based SXP are used to advertise IP-SGT mappings across the devices.

The main purpose of SGT is policy enforcement. The policies are defined in Cisco ISE as SGT ACLs (SGACL), and are based on source and destination tags. As the SGACLs are defined in ISE, meaning all devices in the network have access to them, and as they are not IP specific, they can be applied anywhere. Access and rights are no longer enforced based on VLANs or IP, but on identity.

SGT are also used to implement segmentation in the network, enabling more security and control.

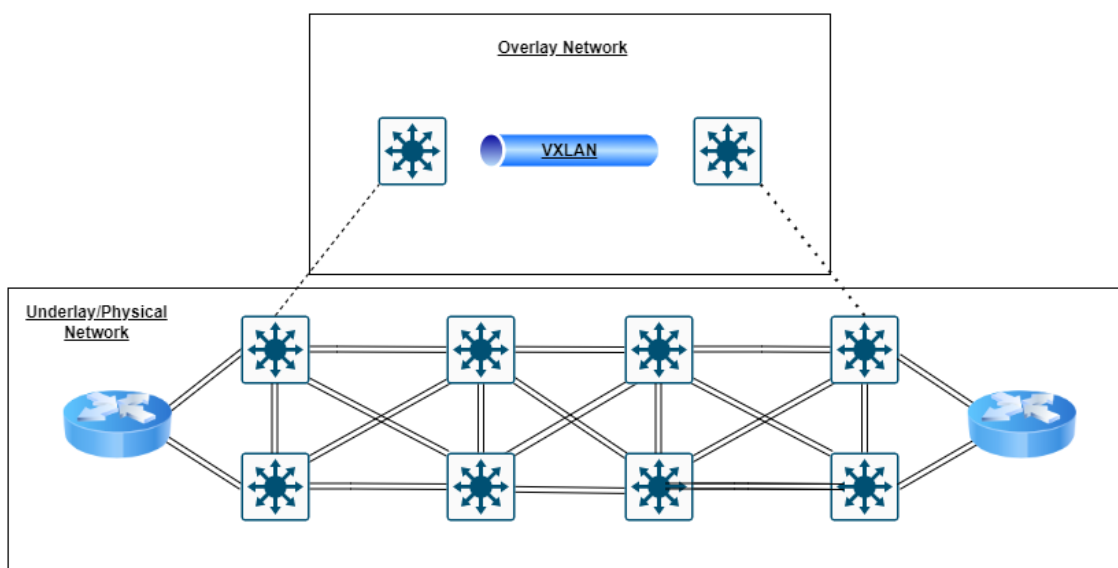
## 2.6.2 Segmentation

In Cisco SDA there are two main ways of segmenting the network; Macro and Micro Segmentation [36].

- *Macro Segmentation*: Macro-segmentation is implemented using Virtual Network (VN), or the traditional network's VRF. Devices that usually communicate are placed in the same VN, and communication between VNs are not permitted.
- *Micro Segmentation*: SGTs are used to enforce micro-segmentation in Cisco SDA. With SGTs policies may be used to permit and restrict access traffic between clients in the same VN. With this second segmentation level, groups in VN "1" can communicate with SGT group "B" in VN "2", while still being separated from SGT "C" and "D" in the same VN.

## 2.6.3 Campus Fabric Architecture

Another important part of Cisco SDA is the Campus Fabric architecture. A fabric is an overlay network, which is a logical topology for virtually connecting devices, laying on top of the underlay network (the traditional, physical network). The overlay network can be both Layer two and three, and multiple fabrics can run on the same underlay network. The Campus Fabric is made up of the Control Plane, Data Plane and Policy Plane. These three planes are represented as Locator/Identifier Separation Protocol (LISP), Virtual Extensible LAN (VXLAN) and Cisco TrustSec [36].

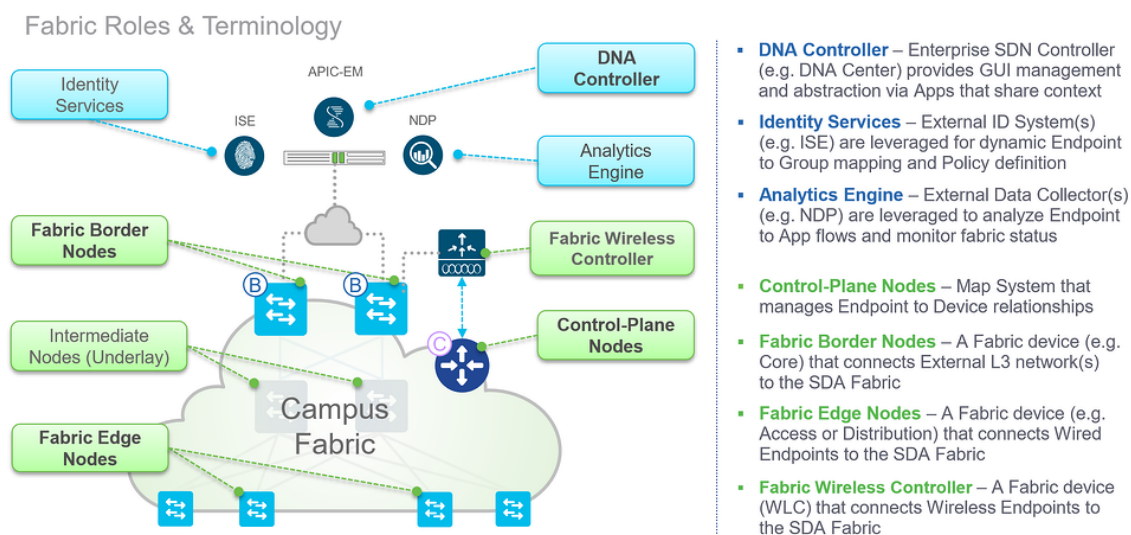


**Figure 2.5:** This figure visualizes the overlay network on top of the physical network.

- *LISP*: Locator/Identifier Separation Protocol is used as the control plane. LISP separates identity from location, unlike the traditional network where location and identity are based on IP addresses. When an endpoint connects to the access device, its IP, MAC and location, also called Routing Locator (RLOC), is registered in a mapping server. The mapping server has the RLOC of the whole network, and when a device wants to send traffic, it requests the RLOC of the destination, thus reducing IP entries in routing tables.

- **VXLAN:** The data plane is based on Virtual Extensible LAN (VXLAN). This is an encapsulation method that allows for lower level data packets to be forwarded across a Layer 3 infrastructure, and is the main component behind the campus fabric [5].
- **Cisco TrustSec:** The policy plane is based on Cisco TrustSec, which uses SGTs to apply policies. (see 2.6.1)

Below is a figure showing an overview of the campus fabric architecture and the different components, together with a short explanation.



**Figure 2.6:** This figure shows an overview over the campus fabric architecture and its roles [37].

## 2.7 Cisco ISE

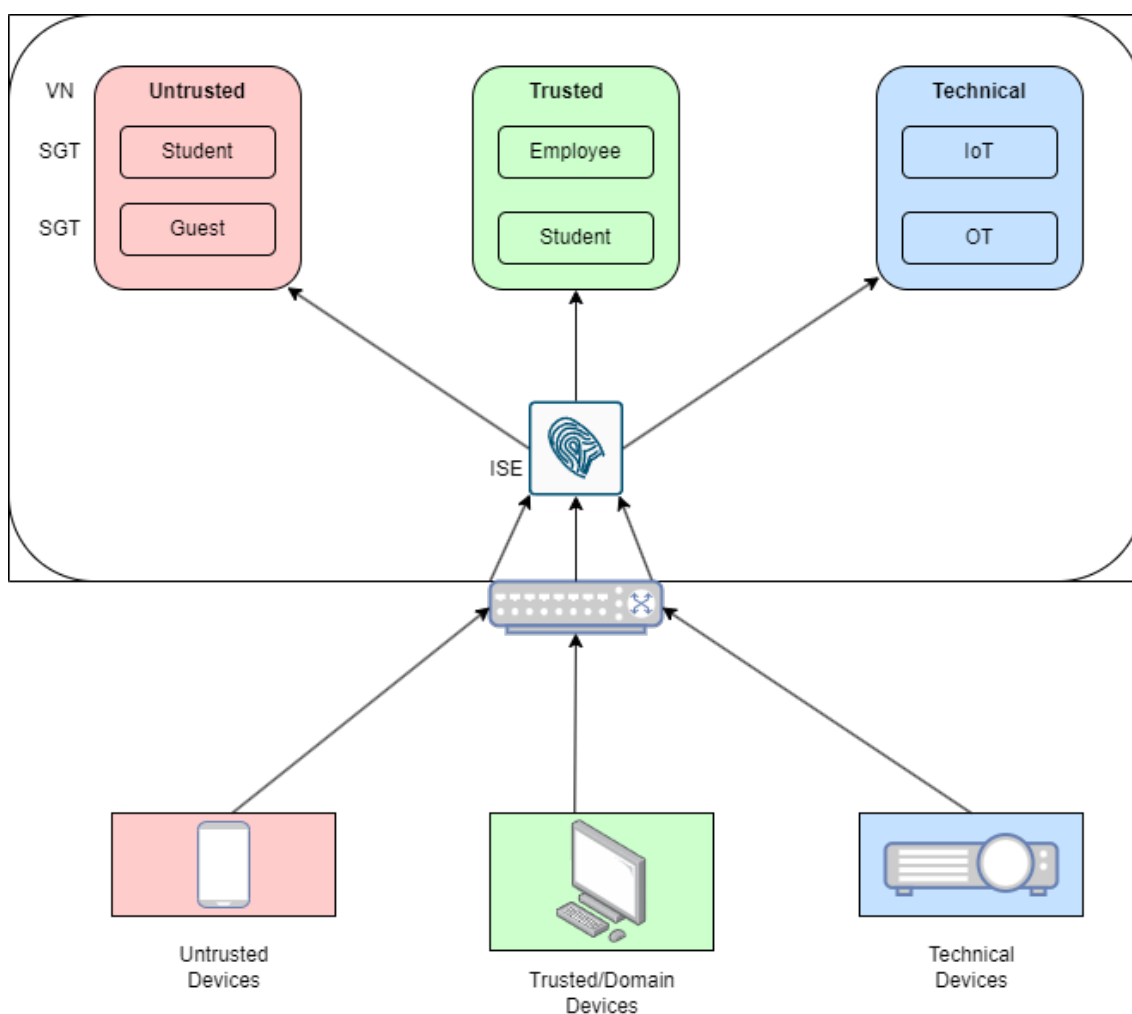
NTNU-IT will be using Cisco Identity Service Engine (ISE) for policy management in the new network. Cisco ISE is an AAA server and a management platform for identity-based network access and policy enforcement. With ISE, real-time data for a network like users, devices access time and location is available and can be used to govern the network. [38].

### 2.7.1 Policies in ISE

At its core, ISE is a policy server where policies, collected in policy sets, are used to filter out requests from network devices. Several criteria and conditions are used, and can be applied at three levels [39]:

- **Policy Set Conditions:** At this policy level, incoming requests are filtered based on conditions like location and device types to sort the traffic before it goes to the next step. The Cisco ISE policies are rule-based, where the rules consist of conditions. There are two condition types [40]:
  - **Simple Condition:** The simple conditions are built up of an operand (attribute), an operator and a value. An example is that the Operating System (OS) should equal Android. Here the OS is the operand, equals is the operator and Android the value that the condition has to match against.

- *Compound Condition*: When two or more conditions are connected by an AND or OR, it is called a Compound Condition.
- *Authentication Policy Conditions*: Authentication conditions are optional, but might be used to authenticate the users against a database, two examples being Active Directory and the internal database in ISE. Should one use multiple identity databases, the order of what database to lookup first can be defined and ISE will go through all until a match is found or none entries matched. The user will be denied access should the authentication fail [40].
- *Authorization Policy Conditions*: The last conditions are used to determine the users' roles and permissions based on for example AD-Groups. Permissions can be collected in an Authorization Profile. The Authorization Policy can consist of one or multiple rules [40].



**Figure 2.7:** This figure visualizes how Cisco ISE, depending on the type of device, places the different devices in different VNs and SGTs.

### 2.7.2 Using MAB with RADIUS and Cisco ISE

Both MAB and RADIUS server are functions included in Cisco ISE [38]. ISE can be configured to work as an RADIUS server, authenticating the devices with MAB. ISE can also be used as the

identity database where all devices that have to be authenticated with MAB can be registered. The devices are registered in *endpoint groups*. These groups are collections of similar types of devices, for example all multimedia devices are registered in one endpoint group and all printers in another. It is then possible to assign permissions based on a device's endpoint group.

When a device wants to authenticate using MAB, ISE working as the RADIUS server, checks the local database of registered devices. If the device is not registered, no access is given. If registered, the device is given an SGT and permissions based on its endpoint group.

### 2.7.3 Cisco ISE APIs

Application Programming Interface (API) is a solution for two computer entities to interact through standardized methods. A common use is to allow software to offer a service or a function to other software or programs without exposing internal functions. Using APIs, an admin can manage end points, users, security groups through code. With HTTP requests the program can receive information or create, update and delete resources, this is often called CRUD. These operations are done through HTTP methods like GET, POST, PUT, PATCH and DELETE [41]:

- *GET*: This method retrieves information from a server. GET is read-only and requires no payload
- *POST*: This method transmits data to a server and is used for creating new resources.
- *PUT*: This method updates an existing resource. When PUT is used, the whole resource is updated with the data provided.
- *PATCH*: This method also updates an existing resource, but can be used to only update some elements of the resource with specific changes. PATCH will not be used for this project
- *DELETE*: This method deletes a resource from the server.

APIs ensure uniform interaction with software, in this case Cisco ISE. This allows automation in form of scripts that can perform management tasks easier with a much higher potential for scaling productivity. ISE provides an API service which includes documentation on REST APIs. The service can be used either as an admin, with full access to read, write, update and delete, or as an operator with only read privileges. Cisco ISE nodes can currently be managed with two formats of APIs.

- *External Restful Services (ERS)*: A widely used format for APIs based on HTTPS and utilizing port 9060.
- *OpenAPI*: An open-source-based format focused on simplicity.

Cisco ISE allows Create, Read, Update, Delete (CRUD) in terms of endpoints, endpoint groups, nodes and policies. Documentation provides the correct format for transmitted data, called the payload. This payload will then be sent to the ISE database to perform its given operation with the contained data. Operations like Read does not need contained data and operations like delete only needs the given ID of the object. However, when creating or updating resources, it is critical that the data is present and formatted correctly for the ISE-server to be able to read and execute the operation.

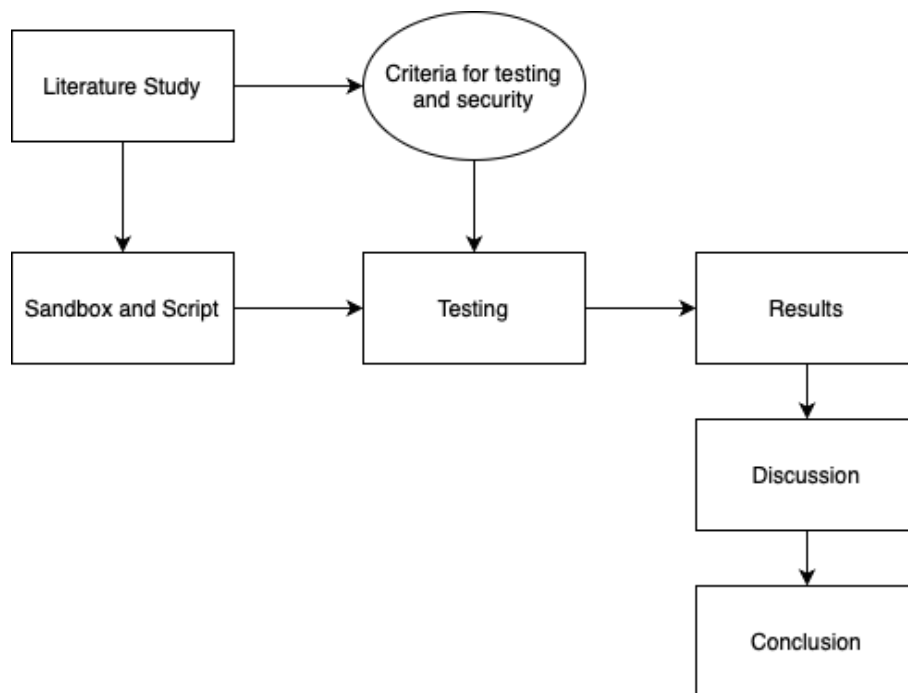


# Chapter 3

## Method

### 3.1 Chapter Outline

This chapter explains the method used to answer the thesis topic and the belonging research questions. The method consists of multiple steps, starting with literature study to research how MAC authentication can be used as a safe mean of access control. From this, criteria for testing and security evaluation were developed together with a test environment consisting of a sandbox and a script. The next step in the method was then to test the criteria in the environment. This produced a result, which will be discussed and concluded in respectively chapter five and six. The flow of this method is shown in the image below (figure 3.1).



**Figure 3.1:** The figure shows the flow of the method used in the project. Starting with Literature Study, which leads to the criteria used for testing. The test environment (sandbox and script) is used in the testing. The testing produces some results, which are then discussed and concluded.

## 3.2 Literature Study - Information Gathering and Research

Literature study is an important part of this project, and many of the results are based on the literature and research articles gathered using this method. This section presents the different steps in the method and criteria used for collecting the sources used in the thesis. This step is used to support all three of the research questions; To what extent can MAC Authentication be used to prevent unauthorized access to wired networks? Are there solutions combined with MAC authentication that could make MAC authentication more secure? What options exist for authenticating devices that do not support 802.1x?

Mainly, the literature study was used to research the topic of MAC authentication, which alternatives exist, and the general consensus regarding the security of MAC authentication. In addition, this part of the method was used to find possible improvements to the MAC authentication and to then use the other steps to test these hypotheses.

### 3.2.1 Search Engine

In researching the topics related to the thesis, research questions and collecting relevant information, the Google Scholar search engine was used. This present the user with research articles related to provided keywords. It is also possible to see the number of times the article has been referred to in other research articles, giving one way of assessing the quality of the article. Google Scholar was used frequently during the research period to gather satisfactory and relevant information to the report. This was also a way to ensure the information presented and being assessed was relevant and had a high academic level.

### 3.2.2 Criteria for Selecting Sources

When gathering relevant sources different criteria has been used to assess whether the source is of the desired standard, and how much available information there is around the topic.

#### Number of References

A satisfactory criteria for the sources, is the number of times the research article has been referenced. Google Search engine gives an overview of how many times the article has been referenced and by which articles, and this has been used when selecting the sources.

#### Year of Publication

The year of publication is also a criteria to assess the source. The project tried to make use of as new sources as possible, to ensure that the documented literature is up to date according to the most recent development. Some systems are well documented and is still used despite being several years old. In these cases the sources has been used, regarding the age of the publication.

#### The Publisher of the Source

Another criteria for the assessment of the literature is whether the source is published by a private individual, an individual through a organization (e.g. a university) or if its published by an organization (e.g. IEEE). Literature published by an organization or through a representative of an organization gives the source more credibility than by an private individual. This is therefore taken in to consideration when selecting literature for the thesis.

### 3.3 Testing

The second step in the method used in this project was testing. After the literature study, a theory had come forth:

The problem with MAB is the fact that the authentication is based on one attribute, the MAC address. By adding more attributes to the authentication process, the process would become more complex and thus working as a security addition. Two attributes were chosen, based on how easy testing the attributes would be. These attributes was location and hardware.

The tests were conducted by making policies in ISE that in theory could help secure MAB, and see if it worked by trying to connect a test device to the network. This step was used to answer research questions one and two; the extent MAC authentication can be used to prevent unauthorized access, and whether there are solutions combined with MAC authentication that could make MAC authentication more secure (see chapter 1.2.1).

The testing part of the method is divided into two cases. The first case attempts to gain access to the network without custom attributes, and the second attempts to gain access to the network with custom attributes. Conditions for these two cases is that the threat actor has acquired the MAC address of a device that is authenticated and has access to a part of the network, that the threat actor then uses to access the network. The threat actor acquiring the MAC address of a device can happen through for example the MAC address being physically displayed on the device, or from a phishing attack.

#### 3.3.1 Sandboxing

The method and technologies used, as well as the script, was developed using the Sandbox as a testing environment. The Sandbox was provided by NTNU IT and contains a simplified version of the technologies and design choices used in their network. The figure below illustrates the topology of the sandbox.

Figure 3.2 depicts NTNU's network divided into five areas. The first consists of NTNU IT's legacy network, which is the network they are currently in the process of phasing out. The second area is named Fusion, and is the core of the network. The other areas are managed from Fusion. The new SDA network is placed in area three, and this is where all tests during the project was done. Area four is the data center network, containing among other things the ISE. Lastly, the firewall is placed separately in area five.

Figure 3.3 below shows an example of type of VNs that could be implemented in the sandbox. Two VNs that were implemented and used for the testing are the *Untrusted VN* and *Guest VN*. All devices trying to connect to the network, but not being able to authenticate, are placed in the guest VN. To access the Untrusted VN, the devices have to be registered in ISE and authenticated. The Untrusted VN was therefore used in this project as the network segment that threat actors should not access.

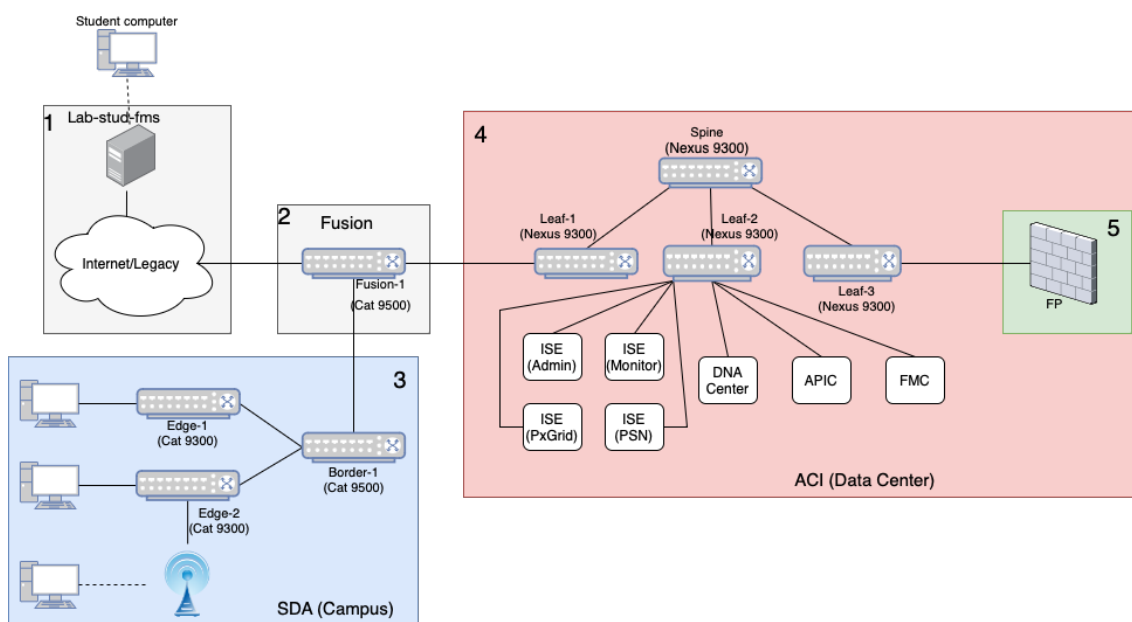


Figure 3.2: The figure shows an overview of the sandbox topology.

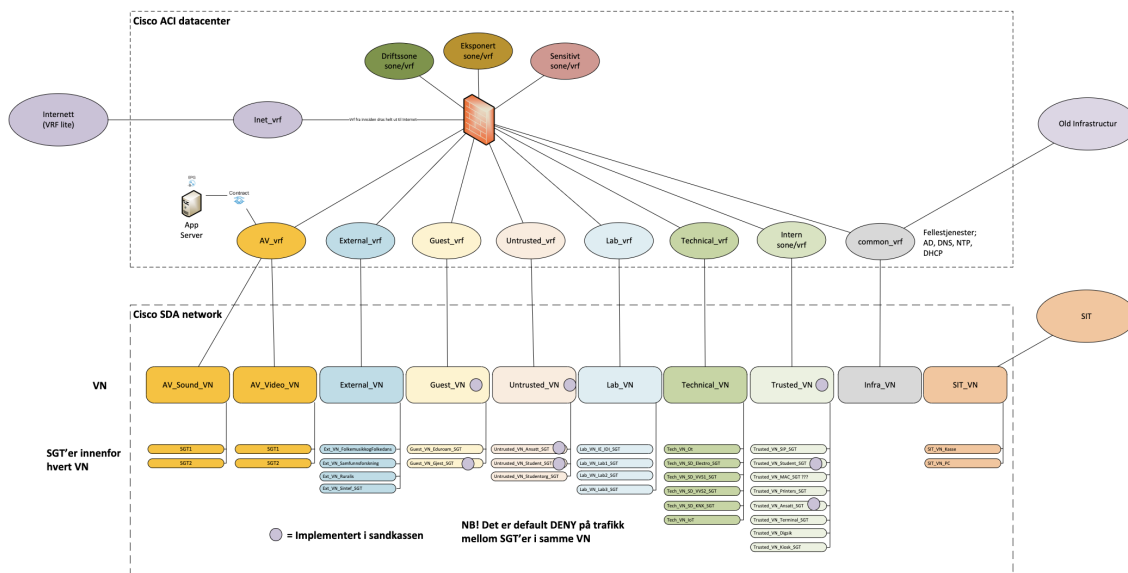


Figure 3.3: The figure shows an overview of VNs in the sandbox. Most of them are yet to be implemented. The VNs used in this thesis was Guest VN and Untrusted VN.

Access to the sandbox was achieved through a VM called "Lab-stud-fms", from a device labeled "Student Computer" in the figure, connected to NTNU IT's legacy network. The traffic flows first through the firewall called "FP" (see number five in 3.2) before accessing the sandbox, and then again before the traffic from the sandbox goes back to the device. As an example, when accessing ISE in the Application Centric Infrastructures (ACIs), the traffic will flow from the "Lab-stud-fms" VM, through the network to the firewall in area five, to ISE in area four, back to the firewall and then back to the VM in area one.

### 3.3.2 Scripting

One part of the thesis was to develop a script for registering devices in ISE. This meant being able to add single devices with default values. Other desired features were deleting and updating already registered devices, as well as adding multiple devices at once and adding devices with custom attributes. The objective of the script was to enable the testing of features for improving the security of MAC authentication.

The script was written in python3 using Cisco API documentation for ISE. Version control was implemented with Git, where changes can be tracked and reverted if necessary. The core operation of the script is the utilization of APIs to achieve the desired functions of the program. This means operating with REST operations and JSON formatted data. The script is a proof of concept and will therefore only contain bare functionality with no Graphical User Interface (GUI) and with little focus on User Experience (UX).

#### Base Functionality

Below is a list of the desired minimum functionalities of the script:

- *Listing Endpoint Information*: For context visibility the script should have a function for listing endpoints and their registered information. This is for checking if a device is registered and if the registered information is correct.
- *Add Endpoint*: Adding endpoints in ISE is the main functionality of the script. The endpoints should be added with minimum the following information:
  1. MAC Address
  2. Endpoint Group
  3. Description
- *Add Endpoints in Bulk*: Often when new devices are to be registered, there are many new devices. The script should therefore have the functionality to add multiple devices from a JSON or Comma-separated Values (CSV) file.
- *Edit Endpoint*: Should an endpoint change, or something went wrong during the initial registration, one should be able to update the endpoint with the newest information.
- *Delete Endpoint*: When a device no longer has the need to be connected to the network, the script should have the functionality to remove said device from ISE.

### **Additional Functionality**

In addition to the functionality listed above, the script should be able to add custom attributes to the endpoint in ISE. The custom attributes was needed test if MAC authentication can be used in a more secure manner.

### **Test of Script's Functionality**

To test if the script worked as intended, all functions of the script would be tested one by one, and improved until all functions worked as desired. This script works as desired if it can be used to register a device, and the device can connect to the network and gain access to the Untrusted VN.

### **3.3.3 Test Case One - Access Attempt Without Custom Attributes**

To create a baseline for further testing to see if any methods can help prevent spoofing, a spoofing attempt was made. The test consists of trying to spoof a MAC address without additional attributes on one of the devices not registered in ISE.

### **Risk Mitigation**

In test case one, nothing has been done to mitigate a successful spoofing attack. If the test is successful the spoofing device should therefore gain access to the *untrusted\_lan*.

### **Prerequisites**

Before the testing could start, some preparatory work had to be done. Policies had first to be set up in ISE to be enforced upon the test traffic. This was done by first separating the test devices and data from the bachelor thesis from the devices and data from NTNU, by creating an endpoint group for these tests called *fms-testgroup*.

The next step was making a policy-set (see chapter 2.7.1), but as NTNU already had an existing policy-set for wired MAB, this was used instead to avoid complications. Since connection-requests are checked against the policy-sets top to bottom, a test policy-set for MAB registered with lower priority would never be used as the production policy-set would capture all incoming MAB-requests. For an incoming request to be checked against the "MAB-Kabel" policy-set, it must match the "Wired\_MAB" condition (see figure 3.4).

Status	Policy Set Name	Description	Conditions	Allowed Protocols / Server Sequence	Hits	Actions	View
+	MAB-Kabel		Wired_MAB	Default Network Access	63095		
+	Default	Default policy set		Default Network Access	268		

**Figure 3.4:** The MAB policy-set containing policies to be enforced on devices accessing the network through MAB.

Inside the policy-set, an Authorization Policy named *fms\_test* was made with a condition checking if the connecting device is registered in the identity group *fms\_testgroup* (see figure 3.5). If the device is registered, it should be given the *Untrusted\_VN\_Student\_SGT*, and now have access to the *untrusted\_lan*. If the device is not registered, it will be given the default SGT *Guest\_VN\_Gjest\_SGT* and access to the *guest\_lan* (see figure 3.6).

✓	fms_test	IdentityGroup-Name EQUALS Endpoint Identity Groups:fms-testgroup
---	----------	--

**Figure 3.5:** This figure shows the policy condition made to check if the device requesting access is registered in the *fms\_testgroup*

✓	fms_test	IdentityGroup-Name EQUALS Endpoint Identity Groups:fms-testgroup	untrusted_lan	Untrusted_VN_Student...	6
✓	Default		guest_lan	Guest_VN_Gjest_SGT	26440

**Figure 3.6:** This figure shows the Identity Group Condition that has to match to gain access to the *Untrusted\_lan* (marked with green). If not, it is set to the default lan (marked with red)

### Testing and Validation Criteria

To test spoofing when no extra attributes are used, to PCs was used. PC 1, a Linux device, spoofed the MAC address of PC 2 before trying to connect to the network. This was done by using the following command on PC 1: `macchanger -m F8-75-A4-84-36-3A eth0`. During the test, PC 2 was never connected to the network, which means that if the MAC address of PC 2 shows as authenticated in the ISE logs, it is in fact PC 1 that is connected. The criteria for a successful spoof is if the devices shows as authenticated on the *Untrusted\_lan*.

### 3.3.4 Test Case Two - Access Attempting With Custom Attributes

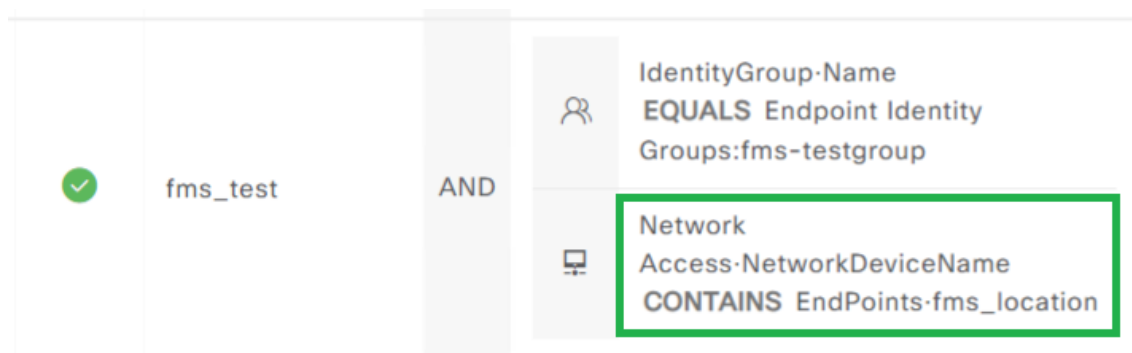
A theory on how MAC authentication can be used to prevent unauthorized access and how to make MAB more secure, was that if a connecting device had to match more than one criteria (MAC address) to authenticate, the chance of a successful spoofing attack could be limited. Two attributes were tested, a location attribute and hardware attribute. This section will first explain the prerequisites for the tests before explaining the actual tests done.

#### Risk Mitigation

The theory of tests of case two is that the risk of a successful spoofing attack will be limited as more attributes has to match to gain access, not just the MAC address. In a successful test the spoofed device should therefore not gain access.

#### Location Attribute Prerequisites

To test the second case, new policies for checking the location of the authenticating device had to be made. For this thesis, the location-policy made was based on the name of the access switch where the device connects. This was possible as the sandbox access switches at NTNU has a structure where the location of the switch is present in the name: *sand-location-sw*. For example *sand-sluppen-3etg-sw10*. A condition was then added to the policy made for the first test case, where the name of the network device has to contain the location registered with the device in ISE.



**Figure 3.7:** A location condition was added so that both the MAC address and registered location have to match to be given access to the *untrusted\_lan*.

#### Hardware Attribute Prerequisites

When a device connects to the network with MAB, ISE automatically detects the hardware manufacturer and the hardware model. The format of the hardware attribute is *type-device*, for example *Lenovo-Device* or *Dell-Device*. A policy checking if the detected hardware matches the registered hardware was made and added to the *fms\_test* policy set.



### Test of Location Attribute

Testing of the location attribute was done in two steps. First the solution of using the switch name as the location attribute had to be tested. This was done by registering a device in ISE with *Sluppen* as its location. The registered device should gain access to the "untrusted\_lan" when connecting.

The next test was done to see if using the location attribute could mitigate spoofing. PC 1 was registered to ISE with *Kalvskinnet* as location, which is a different location than where the testing was done. PC 2 was then used to spoof the MAC address of PC 1, and then tried to connect to the network from Sluppen. The expected result is that the authentication should fail, and the PC should be put on the guest LAN.

### Test of Hardware Attribute

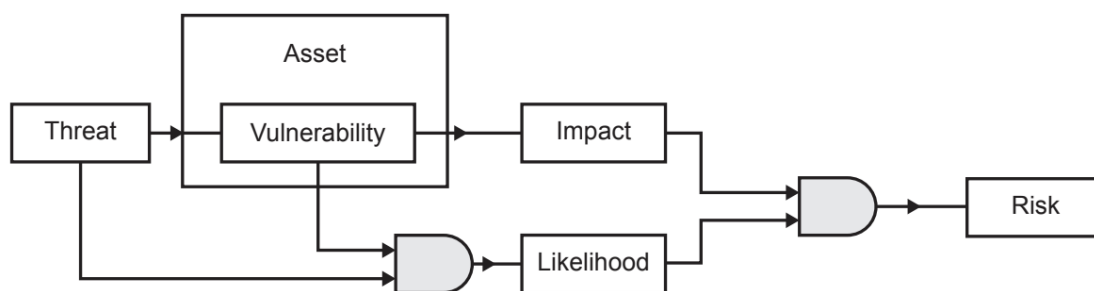
Two tests were conducted to see if using the hardware attribute could be used to make MAB more secure. The first test checked if the hardware attribute is based on the vendor part of the MAC address. The second test was conducted to see if the attribute could prevent access from a spoofing attack, but this was only to be done if the first test showed that the hardware was not derived from the MAC address.

To check if the device type is based on the MAC address, a Dell PC was used to spoof the MAC address of a MacBook (C8:89:F3:B5:81:5A) with the following command: `sudo macchanger -m F8-75-A4-84-36-3A eth0`. When the Dell connects to the network, it is possible to gather the hardware details from the ISE live log. If the logs showed that a MacBook tried to connect to the network, the hardware details are based on the MAC address. Should the hardware details not be based on the MAC address, further testing could be done to see if the attribute could help mitigate a spoofing attack.

## 3.4 Criteria for Security Evaluation

The thesis question concerns the security of an authentication solution consisting of MAB and alternatives for this solution. Assessment criteria regarding both MAB and the alternatives is important in order to have a good assessment basis. Based on these criteria the solutions can be classified as an improved version of solely MAB.

The threat's consequence and the event's probability are considered when assessing the risk. This risk is defined in the ISO standard as "the effect of uncertainty on objectives" (ISO Guide 73:2009) Sutton [42]. In the risk environment (depicted below in figure 3.8), the risk is based on different attributes: threat, vulnerability, impact and likelihood.



**Figure 3.8:** The figure shows an overview of the risk environment [42].

The criteria used to define if a solution is more secure than solely MAB are based on the attributes that define the risk. By reducing these attributes the risk will also be reduced, and these additions to the solution will be considered an improvement. These criteria are described in the paragraphs below.

#### **Lowens Chance of Exploitation - The Likelihood**

The baseline for security in this thesis is an open port without any added security. In this scenario exploitation through MAC spoofing is quite simple. Based on a case where physical access is accomplished, all that is needed is the MAC address of the original device (see chapter 2.4) to spoof the device and connect to the port.

For a solution to be considered an improvement, it has to stop or increase the complexity of the threat actor process for breaching the network. By increasing the complexity that an attack must have in order to succeed, the likelihood will also decrease. An improvement to MAB will provide this increased complexity. The likelihood of an attack is also dependent on the probability of the attacker being detected in the system, and if there are other controllers implemented to prevent and handle a possible attack. Contributing with this to MAB will also be seen as an improved solution [42].

#### **Lowens Consequences of Exploitation - The Impact**

It is important that in the case of an exploitation, the consequences ends up being as small as possible. By reducing the consequences of a exploitation, the risk of that asset can also be reduced, as indicated in figure 3.8 above. A solution that minimizes the attack surface will be considered an improvement of the original solution. In addition, implementing methods for monitoring and detecting incidents early in the process can help mitigating the negative impacts. Preparing for and knowing how to handle an incident can shorten the incident handling, making it possible to act as fast as possible and mitigate the impact of the incident.

#### **Ease of Use**

In addition to the attributes mentioned in the risk environment, ease of use is also important for a good solution. For a solution to be viable, it has to be possible to manage it in a network at the scale of NTNU. Even though a solution greatly improves the security of the solution, if the execution requires resources NTNU does not have or cannot spare, the solution will not be categorized as an improvement.

**The Cost**

The resources required to make use of a proposed solution is also a criteria. The solution has to be attainable in terms of time used on the solution, economic cost and the resources used by the solution for it to be a good addition to MAB.

# Chapter 4

## Results

This chapter presents the results from completed tests based on the method described in chapter 3. Firstly the results from developing the script are presented, divided into the base functionality of the script and the added functionality to create custom attributes. After that, the results from the testing are presented, first from testing without any custom attributes and then lastly with the use of custom attributes.

### 4.1 Testing Prerequisites

#### 4.1.1 Sandboxing

The sandbox as a test environment worked as expected. Connection to the test environment was easily done with SSH to the student test server. With this testing environment we were able to verify the success of functions like adding, updating and removing endpoints. The sandbox also enabled the testing of added security features like location attributes and ISE policies.

#### 4.1.2 Scripting

As explained in the method, the goal of the script was to perform administrative operations in Cisco ISE and to enable testing of security features. The script was able to perform its primary function of adding, updating and deleting endpoints using Cisco ISE APIs. Functionality was also implemented for adding endpoints in bulk, using a JSON-formatted file, allowing for a larger degree of automation in endpoint operations.

The script is formatted using a main interface script with function calls to other scripts. The segmentation of functions is based on available functionality in OpenAPI and Cisco ERS. OpenAPI was used as much as possible, but we encountered challenges in the form of limited functionality in OpenAPI. Therefore, for this script, OpenAPI was only used to add endpoints individually and in bulk. These functions are located in individual python files for a less complicated code structure. Functionality not available in OpenAPI is located in a different python file which uses Cisco ERS to perform operations like updating, deleting and listing endpoints by certain criteria. The script can be found in Appendix A.

```
Cisco ISE Endpoint manager

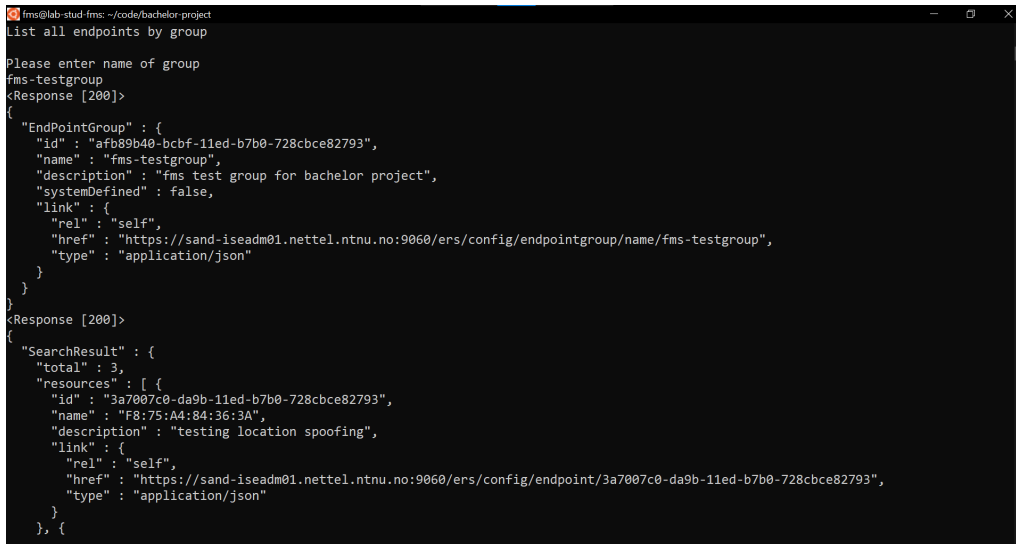
Press 1 to list all endpoints in group
Press 2 to display single endpoint by MAC
Press 3 to create endpoint
Press 4 to update endpoint
Press 5 to delete endpoint
Press 6 to add endpoints in bulk
Press any other key to exit
```

**Figure 4.1:** This picture shows the main interface of the script with the available choices for selection.

## Base Functionality

### Listing Endpoint Information

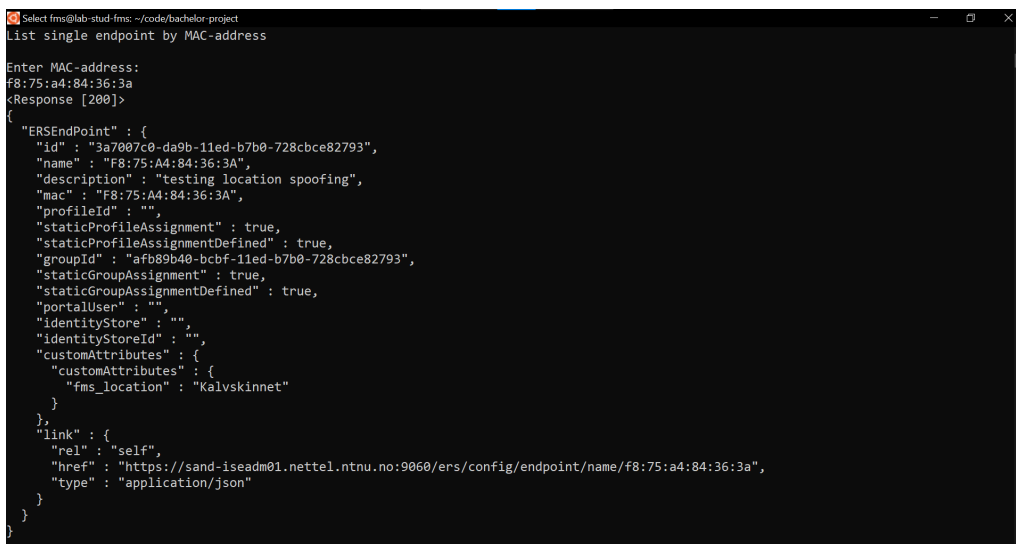
The figure below shows how one with the script can list all endpoints in an endpoint group in ISE. The second figure shows that the script can be used to list all the information of one endpoint based on its MAC address.



```
fms@lab-stud-fms: ~/code/bachelor-project
list all endpoints by group

Please enter name of group
fms-testgroup
<Response [200]>
{
  "EndPointGroup": {
    "id": "afb89b40-bcbf-11ed-b7b0-728cbce82793",
    "name": "fms-testgroup",
    "description": "fms test group for bachelor project",
    "systemDefined": false,
    "link": {
      "rel": "self",
      "href": "https://sand-iseadm01.nettel.ntnu.no:9060/ers/config/endpointgroup/name/fms-testgroup",
      "type": "application/json"
    }
  }
}
<Response [200]>
{
  "SearchResult": {
    "total": 3,
    "resources": [ [
      {
        "id": "3a7007c0-da9b-11ed-b7b0-728cbce82793",
        "name": "F8:75:A4:84:36:3A",
        "description": "testing location spoofing",
        "link": {
          "rel": "self",
          "href": "https://sand-iseadm01.nettel.ntnu.no:9060/ers/config/endpoint/3a7007c0-da9b-11ed-b7b0-728cbce82793",
          "type": "application/json"
        }
      }
    ]
  }
}, {
```

**Figure 4.2:** The figure shows that this function can be used to list information about an endpoint group by entering the name of the group. The information includes the registered devices in the group.



```
Select fms@lab-stud-fms: ~/code/bachelor-project
List single endpoint by MAC-address

Enter MAC-address:
F8:75:A4:84:36:3a
<Response [200]>
{
  "ERSEndPoint": {
    "id": "3a7007c0-da9b-11ed-b7b0-728cbce82793",
    "name": "F8:75:A4:84:36:3A",
    "description": "testing location spoofing",
    "mac": "F8:75:A4:84:36:3A",
    "profileId": "",
    "staticProfileAssignment": true,
    "staticProfileAssignmentDefined": true,
    "groupId": "afb89b40-bcbf-11ed-b7b0-728cbce82793",
    "staticGroupAssignment": true,
    "staticGroupAssignmentDefined": true,
    "portalUser": "",
    "identityStore": "",
    "identityStoreId": "",
    "customAttributes": {
      "customAttributes": {
        "fms_location": "Kalvskinnet"
      }
    }
  },
  "link": {
    "rel": "self",
    "href": "https://sand-iseadm01.nettel.ntnu.no:9060/ers/config/endpoint/name/f8:75:a4:84:36:3a",
    "type": "application/json"
  }
}
}
```

**Figure 4.3:** By entering the MAC address of a device registered in ISE, this function can be used to view all registered information of this device.

Based on these results, the script's base functionality of listing endpoints and information worked as desired.

### Add Endpoint

The figures below show that the script was successfully used to add endpoints to ISE. The first figure shows how the script can be used to add an endpoint, where a user must submit the device's MAC address, endpoint group, description and location. The parts of the function regarding location are optional for the base functionality of adding an endpoint.

```

create endpoint
Please enter mac address
f8:75:a4:84:36:3a
Please enter the name of the group for the endpoint
fms-testgroup
Response [200]
{
  "EndPointGroup" : {
    "id" : "afb89b40-bcbf-11ed-b7b0-728cbce82793",
    "name" : "fms-testgroup",
    "description" : "fms test group for bachelor project",
    "systemDefined" : false,
    "link" : {
      "rel" : "self",
      "href" : "https://sand-iseadm01.nettel.ntnu.no:9060/ers/config/endpointgroup/name/fms-testgroup",
      "type" : "application/json"
    }
  }
}
Please enter description of endpoint
Testing script with adding of endpoint
Please enter location
kalvskimmet
{"groupId": "afb89b40-bcbf-11ed-b7b0-728cbce82793", "description": "Testing script with adding of endpoint", "mac": "f8:75:a4:84:36:3a", "name": "f8:75:a4:84:36:3a", "staticGroupAssignment": true, "staticProfileAssignment": true, "customAttributes": {"fms_location": "kalvskimmet"}}
CREATED

```

**Figure 4.4:** The figure shows that a user has chosen the "Create Endpoint" option and needs to submit a MAC address, endpoint group name, description and location to register a device in ISE with the script.

Endpoint Identity Group List > fms-testgroup

Endpoint Identity Group

\* Name **fms-testgroup**

Description

Parent Group

Identity Group Endpoints Selected 0 Total 4

	MAC Address	Static Group Assignment	Endpoint Profile
<input type="checkbox"/>	00:AA:00:BB:00:CC	true	Unknown
<input type="checkbox"/>	01:23:45:67:89:AB	true	Unknown
<input type="checkbox"/>	AA:BB:AA:BB:AA:BB	true	Unknown
<input type="checkbox"/>	F8:CA:B8:21:FB:EE	true	Unknown

**Figure 4.5:** This figure shows the Endpoint Group *fms-testgroup* in ISE. It is populated with endpoints with help from the script, seen at the bottom as a list of MAC addresses.

Figure 4.6 displays that a device trying to connect to the network without being registered in ISE ends up on the *guest\_lan*. After the device is added (marked in figure 4.5 with green), the device gained access to the *untrusted\_lan*, which was the desired result.

Time	Status	Details	Repea...	Identity	Endpoint ID	Endpoint...	Authenti...	Authoriz...	Authorization Pr...	IP Address
Mar 22, 2023 02:26:01.9...	●	🔒	1	F8:CA:BB:21:FB:EE	F8:CA:BB:21:FB:...	Dell-Device	MAB-Kab...	MAB-Kab...	untrusted_lan	10.26.146.253.20 ...
Mar 22, 2023 01:25:51.3...	✓	🔒		F8:CA:BB:21:FB:EE	F8:CA:BB:21:FB:...	Dell-Device	MAB-Kab...	MAB-Kab...	untrusted_lan	fe80::b3af:44ed: ...
Mar 22, 2023 01:24:34.9...	✓	🔒		F8:CA:BB:21:FB:EE	F8:CA:BB:21:FB:...	Dell-Device	MAB-Kab...	MAB-Kab...	untrusted_lan	fe80::b3af:44ed: ...
Mar 22, 2023 12:58:37.0...	✓	🔒		F8:CA:BB:21:FB:EE	F8:CA:BB:21:FB:...	Dell-Device	MAB-Kab...	MAB-Kab...	guest_lan	fe80::b3af:44ed: ...
Mar 22, 2023 12:57:09.0...	✓	🔒		F8:CA:BB:21:FB:EE	F8:CA:BB:21:FB:...	Dell-Device	MAB-Kab...	MAB-Kab...	guest_lan	fe80::b3af:44ed: ...

Last Updated: Wed Mar 22 2023 14:31:23 GMT+0100 (Central European Standard Time) Records Shown: 5

**Figure 4.6:** The two columns marked with black shows the time of the authentication and the MAC address of the connected device. The first two entries (time 12:57 and 12:58) shows that the authentication failed and the device was only given access to the default lan - *guest\_lan*. The device was then registered in ISE, and the three following entries show that the device now gains access to the *untrusted\_lan*.

During the tests of the script it was discovered that the required "name" field got overwritten after an endpoint was registered. If someone registered an endpoint with a name "pc1", the name field would afterwards contain the MAC address of the device (see figure 4.7 below). The registered name, "pc1", was nowhere to be found. The reason why was not discovered.

```
Select fms@lab-stud-fms: ~/code/bachelor-project
List single endpoint by MAC-address

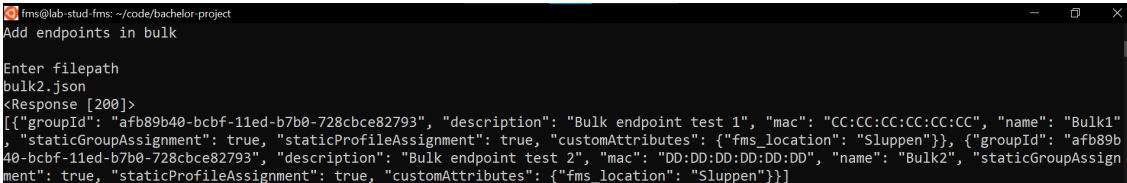
Enter MAC-address:
f8:75:a4:84:36:3a
<Response [200]>
{
  "ERSEndPoint" : {
    "id" : "3a7007c0-da9b-11ed-b7b0-728cbce82793",
    "name" : "F8:75:A4:84:36:3A",
    "description" : "testing location spoofing",
    "mac" : "F8:75:A4:84:36:3A",
    "profileId" : "",
    "staticProfileAssignment" : true,
    "staticProfileAssignmentDefined" : true,
    "groupId" : "afb89b40-bcbf-11ed-b7b0-728cbce82793",
    "staticGroupAssignment" : true,
    "staticGroupAssignmentDefined" : true,
    "portalUser" : "",
    "identityStore" : "",
    "identityStoreId" : "",
    "customAttributes" : {
      "customAttributes" : {
        "fms_location" : "Kalvskinnet"
      }
    }
  },
  "link" : {
    "rel" : "self",
    "href" : "https://sand-iseadm01.nettel.ntnu.no:9060/ers/config/endpoint/name/f8:75:a4:84:36:3a",
    "type" : "application/json"
  }
}
```

**Figure 4.7:** When one lists information on a registered endpoint, one can see (marked with a green box), that the registered name in the name field is overwritten by the MAC address.



### Add Endpoints in Bulk

Multiple endpoints could be added at once from a JSON file. The figure below shows that multiple endpoints are successfully added with the script.



```
fms@lab-stud-fms: ~/code/bachelor-project
Add endpoints in bulk

Enter filepath
bulk2.json
<Response [200]>
[{"groupId": "afb89b40-bcbf-11ed-b7b0-728cbce82793", "description": "Bulk endpoint test 1", "mac": "CC:CC:CC:CC:CC:CC", "name": "Bulk1", "staticGroupAssignment": true, "staticProfileAssignment": true, "customAttributes": {"fms_location": "Sluppen"}}, {"groupId": "afb89b40-bcbf-11ed-b7b0-728cbce82793", "description": "Bulk endpoint test 2", "mac": "DD:DD:DD:DD:DD:DD", "name": "Bulk2", "staticGroupAssignment": true, "staticProfileAssignment": true, "customAttributes": {"fms_location": "Sluppen"}}]
```

**Figure 4.8:** This figure shows that two endpoints called "Bulk1" and "Bulk2" are added simultaneously to ISE with the script. This returns code 200 which means success.

### Edit Endpoint

The script could be used to edit already registered endpoints in ISE. However, as one can see in the figure below, the user has to enter all the device information again, not just edit one attribute. This is because the function had to use the HTTP PUT request instead of the PATCH request. However, it is possible to edit an entry, which means that the edit functionality of the script works to some extent.



```
fms@lab-stud-fms: ~/code/bachelor-project
Edit endpoint
Enter endpoint MAC:
f8:75:a4:84:36:3a
<Response [200]>
{
  "ERSEndPoint": {
    "id": "384a81f0-f300-11ed-b7b0-728cbce82793",
    "name": "F8:75:A4:84:36:3A",
    "description": "Testing script with adding of endpoint",
    "mac": "F8:75:A4:84:36:3A",
    "profileId": "",
    "staticProfileAssignment": true,
    "staticProfileAssignmentDefined": true,
    "groupId": "afb89b40-bcbf-11ed-b7b0-728cbce82793",
    "staticGroupAssignment": true,
    "staticGroupAssignmentDefined": true,
    "portalUser": "",
    "identityStore": "",
    "identityStoreId": "",
    "customAttributes": {
      "customAttributes": {
        "fms_location": "Kalvskinnet"
      }
    }
  },
  "link": {
    "rel": "self",
    "href": "https://sand-iseadm01.nettel.ntnu.no:9060/ers/config/endpoint/name/f8:75:a4:84:36:3a",
    "type": "application/json"
  }
}

Enter group name for endpoint
fms-testgroup
<Response [200]>
{
  "EndPointGroup": {
    "id": "afb89b40-bcbf-11ed-b7b0-728cbce82793",
    "name": "fms-testgroup",
    "description": "fms test group for bachelor project",
    "systemDefined": false,
    "link": {
      "rel": "self",
      "href": "https://sand-iseadm01.nettel.ntnu.no:9060/ers/config/endpointgroup/name/fms-testgroup",
      "type": "application/json"
    }
  }
}

afb89b40-bcbf-11ed-b7b0-728cbce82793
Enter new endpoint description:
Testing updating endpoint
Enter new location:
Sluppen
<Content-type: 'application/json', 'accept': 'application/json', 'cache-control': 'no-cache', 'Connection': 'close'>
<Response [200]>
{
  "UpdatedFieldList": {
    "updatedField": [
      {
        "field": "description",
        "oldValue": "Testing script with adding of endpoint",
        "newValue": "Testing updating endpoint"
      },
      {
        "field": "customAttributes",
        "oldValue": {"fms_location": "Kalvskinnet"},
        "newValue": {"fms_location": "Sluppen"}
      }
    ]
  }
}

{"ERSEndPoint": {"groupId": "afb89b40-bcbf-11ed-b7b0-728cbce82793", "description": "Testing updating endpoint", "mac": "f8:75:a4:84:36:3a", "name": "f8:75:a4:84:36:3a", "staticGroupAssignment": true, "staticProfileAssignment": true, "customAttributes": {"customAttributes": {"fms_location": "Sluppen"}}}]
```

**Figure 4.9:** This figure shows the script updating the location and description of endpoint.

### Delete Endpoint

The last functionality that was successfully implemented was deleting a given endpoint. An example is shown in the figure below.

```

Delete endpoint
Enter endpoint for deletion
f8:75:a4:84:36:3a
Are you sure you want to delete MAC-address: f8:75:a4:84:36:3a?
(response [200])
{
  "ERSendPoint" : {
    "id" : "3a7007c0-da0b-11ed-b7b0-728cbe82793",
    "name" : "f8:75:a4:84:36:3a",
    "description" : "Testing update of endpoint",
    "mac" : "f8:75:a4:84:36:3a",
    "profileId" : "",
    "staticProfileAssignment" : true,
    "staticProfileAssignmentDefined" : true,
    "groupId" : "afb89b40-bcbf-11ed-b7b0-728cbe82793",
    "staticGroupAssignment" : true,
    "staticGroupAssignmentDefined" : true,
    "portalUser" : "",
    "identityStore" : "",
    "identityStoreId" : "",
    "customAttributes" : {
      "customAttributes" : {
        "fms_location" : "Kalvskinnet"
      }
    }
  },
  "link" : {
    "rel" : "self",
    "href" : "https://sand-issadm01.nettel.ntnu.no:9060/ers/config/endpoint/name/f8:75:a4:84:36:3a",
    "type" : "application/json"
  }
}
(response [204])

```

**Figure 4.10:** This figure shows the script deleting an endpoint and responding with the code 204, which means success.

### Additional Functionality

When the base functionality was ready and working, additional functionality was added to support testing using custom attributes to secure MAB further. As the figure below shows, the custom-made attribute *fms\_location* was successfully added to Cisco ISE when registering a new device.

General Attributes		Custom Attributes	Other Attributes
	Attribute String	Attribute Value	
	fms_location	Kalvskinnet	

**Figure 4.11:** The figure shows that the script could be used to add endpoints with the custom-made attribute *fms\_location* to ISE.

## 4.2 Case One: Access Attempt Without Custom Attributes

To show how easy spoofing a MAC address is and gain access to the network with MAB as an authentication method, a spoofing attempt was made. Figure 4.12 shows the original MAC address of the Linux device used for the spoofing. Figure 4.13 shows that the command was successfully used for changing the device's MAC.

```
(root@kali)~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
    inet6 fe80::a00:27ff:fe0e:348d prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:0e:34:8d txqueuelen 1000 (Ethernet)
    RX packets 1 bytes 590 (590.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 14 bytes 1328 (1.2 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

**Figure 4.12:** The original MAC address of the Linux device is marked with the green box in the figure.

```
(root@kali)~# macchanger -m f8:75:a4:84:36:3a eth0
Current MAC: 08:00:27:0e:34:8d (CADMUS COMPUTER SYSTEMS)
Permanent MAC: 08:00:27:0e:34:8d (CADMUS COMPUTER SYSTEMS)
New MAC: f8:75:a4:84:36:3a (unknown)
```

**Figure 4.13:** The *macchanger* command has been run in the screenshot above, and the MAC address of the Linux device was successfully changed, as seen in the last line "New MAC:".

As the device that was spoofed during this test never was connected to the network, it is the Linux device with the spoofed MAC address of *F8-75-A4-84-36-3A* that is displayed in figure 4.14 below. It is successfully authenticated and gained access to the *untrusted\_lan*, even though the device itself is not registered in ISE. The test was therefore successful and worked as theorized.

## Cisco ISE

### Overview

Event	5200 Authentication succeeded
Username	F8:75:A4:84:36:3A
Endpoint Id	F8:75:A4:84:36:3A
Endpoint Profile	Unknown
Authentication Policy	MAB-Kabel >> Default
Authorization Policy	MAB-Kabel >> fms_test
Authorization Result	untrusted_lan

### Authentication Details

Source Timestamp	2023-04-14 09:35:50.695
------------------	-------------------------

**Figure 4.14:** This log entry shows that a device with MAC address *F8:75:A4:36:3A* is authenticated and got access to *untrusted\_lan*. As the device whose MAC address is never connected to the network, this shows that the MAC address in the screenshot is the spoofed MAC address.

## 4.3 Case Two: Access Attempting With Custom Attributes

As the results showed from the baseline test in section 4.2, spoofing a device can easily be done. The tests in case two used custom attributes and the policies in ISE to help answer thesis question one and two; if MAC authentication can prevent unauthorized access and if there are solutions to further secure MAC authentication.

### 4.3.1 Location Attribute

This section presents the results from using the location attribute, which was done to see if the attribute can be used to mitigate spoofing. A Lenovo PC (F8-75-A4-84-36-3A) was registered with "Kalvskinnet" as the location (see figure 4.15). As the result shows in figure 4.16 when the Dell PC spoofed the Lenovo's MAC address from "Sluppen", it did not gain access to the "untrusted\_lan" as it did in chapter 4.2.

**Cisco ISE**

Username: **f875a484363a**

Endpoint Profile: Unknown

Current IP Address:

Location: Location → All Locations

Applications **Attributes** Authentication

General Attributes **Custom Attributes** Other Attributes

Attribute String	Attribute Value
×	Attribute String
	Attribute Value
fms_location	<b>Kalvskinnet</b>

**Figure 4.15:** The figure shows the registered location attribute ("Kalvskinnet") for the device with MAC address *F8-75-A4-84-36-3A* that needs to match when the device wants to authenticate on the network.

## Cisco ISE

### Overview

Event	5200 Authentication succeeded
Username	F8:75:A4:84:36:3A
Endpoint Id	F8:75:A4:84:36:3A
Endpoint Profile	Unknown
Authentication Policy	MAB-Kabel >> Default
Authorization Policy	MAB-Kabel >> Default
Authorization Result	guest_lan

### Authentication Details

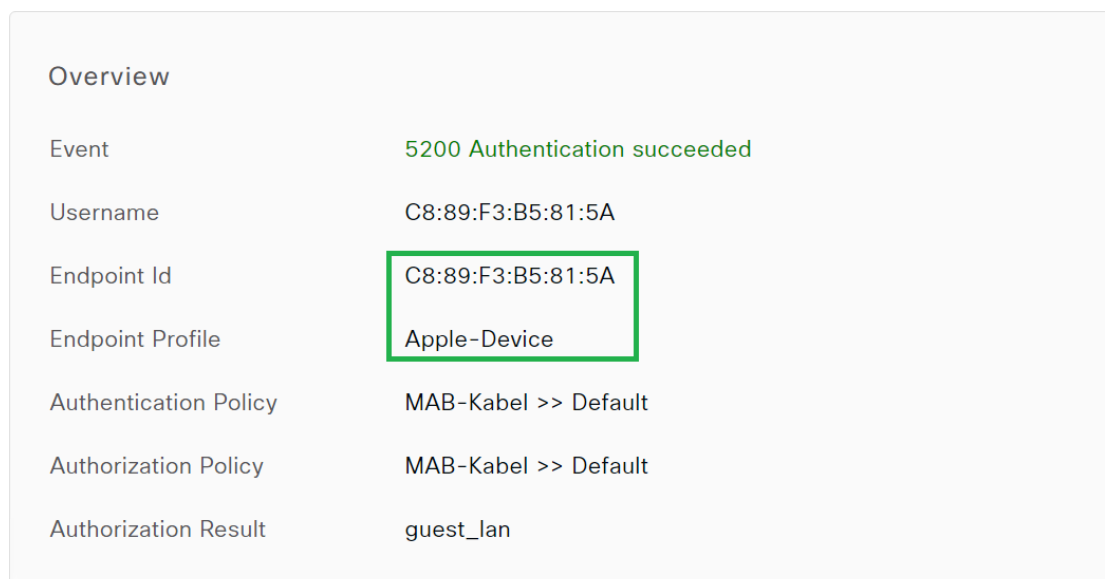
Source Timestamp	2023-04-14 10:08:52.469
------------------	-------------------------

**Figure 4.16:** The device that connected with the MAC address of *F8-75-A4-84-36-3A* in the figure was the malicious PC. The malicious PC got access to the *guest\_lan*, marked with green, which shows that the access attempt failed when the location of the device did not match the registered location.

### 4.3.2 Hardware Attribute

To test if the hardware attribute could be used for testing, a Dell device spoofed an Apple device. As figure 4.17 below shows, the Dell showed up as an Apple Device in ISE. Unfortunately, the theory that the hardware was extracted from the vendor part of the MAC address was true, meaning it would not help mitigate a spoofing attack. Further testing was therefore not done.

#### Cisco ISE



The screenshot displays a log entry in the Cisco ISE interface. The entry is titled 'Overview' and shows the following details:

Field	Value
Event	5200 Authentication succeeded
Username	C8:89:F3:B5:81:5A
Endpoint Id	C8:89:F3:B5:81:5A
Endpoint Profile	Apple-Device
Authentication Policy	MAB-Kabel >> Default
Authorization Policy	MAB-Kabel >> Default
Authorization Result	guest_lan

The 'Endpoint Id' and 'Endpoint Profile' fields are highlighted with a green box, indicating that the MAC address was used to identify the device as an Apple device.

**Figure 4.17:** The picture shows a screenshot of the log entry from when the Dell PC connected to the network with the spoofed MAC address of a MacBook. One can see the address and the hardware registered inside the green box. Based on this, ISE uses the MAC address for classifying hardware.

# Chapter 5

## Discussion

In this chapter, the results described in chapter 4 are discussed in the context of the thesis topic, which is to explore secure access to wired networks, as well as the research questions. The goal is to answer these questions and reflect on security in network access within the project's scope.

### 5.1 Literature Provider and Year of Publication

Literature study was used to explore different solutions of how to secure MAB. When considering these solutions, it is important to consider the nature of the source material, including biased opinions and the year of publication.

Cisco is a prominent network technology provider with a lot of documentation and educational material regarding subjects within network and security. Since Cisco is a provider of network gear and systems the information provided might have a weakness to be biased. However, Cisco is a recognized and credible organization and because of the well documented concepts, Cisco's documentation has been extensively used in this thesis. This especially applies when finding documentation about Cisco specific concepts like ISE.

Much of the source material used in the literature study also has a publication date not from recent years. A lot can change over the years and this can impact the relevance of the material. Even with the publication date in mind, some reports with older publication date has been used and it was difficult to find newer material. Based on the research, the functionality and concepts seems stagnant, but some new and incomplete methods for securing RADIUS is presented. An example of this is in this RFC document from 2012, which presents a relatively new and experimental method for securing RADIUS [43]. This is one of the more recent documentation found within the field and can still be considered older material. With this in mind, the thesis has used many of these relatively older sources in the research.

### 5.2 The Script

A part of the project was making a script for registering endpoints in ISE so devices could be authenticated with MAB. The script will after the project be handed over to NTNU IT so they can use this as a prototype for a management tool in the future.



The script was extensively used during the tests to register test devices and delete or update them with the custom-made attributes. Together with the results presented, which show how the script was successfully used, it is safe to say that the script works as intended and should work as a viable prototype. Good network management is an important part of overall network security. The management tool based on the script will enable a safe and effective way of registering the devices that use MAB. In addition, using the script for registering new devices, should some registrations contain any errors, will make it easier to correct this.

The script can be seen in connection with research question two, which is about finding solutions to improve MAC authentication, as good management can decrease human errors, thus helping in making a secure environment where MAB is used. For example, several endpoints can be added using a list, which may have a lower chance of containing incorrect data like a wrong MAC address than if all the addresses were manually typed. The bulk add feature also enables a higher degree of automation which can be built on further to increase ease of scalability in a network.

A disadvantage with the script is the differing API architectures used in the code. The use of both openAPI and ERS contributes to the unnatural segmentation of code functionality in separate files using different programming logic. If the functionality of openAPI increases in the future, the script should be consequentially rewritten to utilize openAPI where possible. This will simplify the script and ease maintenance.

Another disadvantage is the reliance of long ID strings to manage objects in ISE. As an example, an ID string for an endpoint is 26 characters long which makes it dependent on retrieving the ID from ISE when you need to perform operations. An alternative here could be to use the MAC address which will be guaranteed unique as ISE does not allow multiple entries for the same address. However, Using the endpoint's name is impossible due to how ISE doesn't register names in the database. This is an anomaly discovered when writing the script. The payload for the POST request requires a field for naming the endpoint, which could be useful for clarity. However, the contents of this field is replaced with the MAC address when the request is made to the server. This means that the address becomes the "name" and the actual name disappears.

### **5.3 Case One - Access Without Custom Attributes**

The first test, which was attempting a spoofing attack, was conducted to create a baseline for further testing. The results from this test showed that it is relatively easy to impersonate another device, provided that the threat actor can get a hold of the address of an authenticated device on the network. This was as expected, and shows that MAB in itself is vulnerable, and is therefore dependent upon other security systems. Case one answers the first research question; what extent MAC authentication can be used to prevent unauthorized access. This result is also consistent with the literature study carried out at the start of the method.

### **5.4 Case Two - Access With Custom Attributes**

To take advantage of Cisco ISE and the use of policies to prevent unauthorized access to wired networks, two attribute types were explored, a custom-made location attribute and Cisco ISE's

hardware attribute. This was done to answer research question two; are there solutions combined with MAC authentication that can make MAC authentication more secure?

#### 5.4.1 Location Attribute

As theorized, the results from test case two showed that an additional attribute having to match prevented connection to the network from a spoofed device. It is not a perfect solution, as a connection attempt with a spoofed address will still be successful should it be performed at the right location. The attack surface is however considerably reduced, thus mitigating the chance of a successful spoofing attack, as explained in section 3.4 in the method. Another positive aspect of using this attribute type is the fact that the packet containing the location is sent from the switch, not the client. This limits the possibility of manipulating the packet to contain the desired location.

An important problem with this solution is the fact that the policy utilizing the location attribute is built with a "CONTAINS" operator. Even though it isn't necessarily a security problem, it might lead to more work on a management level as the solution is prone to human error. Should, as an example, the one registering the device to ISE misspell the desired location, the device will not gain access to the intended security group as the policy will not find a match. This can be prevented by having the script check the spelling, or the location is chosen from a predefined list. The problem can also occur with an error in the opposite part of the network with the switch being configured with the wrong name. Another problem with the "CONTAINS" operator is that it can match on a location containing another location in its name. Now all devices on one location have access to two locations.

Cisco ISE does however have an actual location functionality available, where network devices can be added to hierarchical, location based groups Cisco [44]. Using these location types would have removed the possibility of human error and the policies matching on wrong locations. But as locations were not yet implemented in the sandbox, and implementing them as a part of the thesis would be too time consuming and out of scope, the custom-made location attribute was used.

As the custom-made location attribute did successfully mitigate a spoofing attack when a connection attempt was done from the wrong location, the solution checks the criteria for lowering the likelihood of a successful spoofing attack. The names of the switches are template based, the script for registering new devices can easily be changed to use new attributes, and policies can be made once and reused. There are no added costs or complicated procedures to make the location attribute work, and the solution will therefore be considered a viable solution for making MAB more secure. This corresponds with the security criteria in 3.4. This also answers research question two, as using the location attribute works as an additional solution to be used with MAB.

#### 5.4.2 Hardware Attribute

The hardware attribute could unfortunately not be used to make MAB more secure, since the results showed that ISE based the hardware attribute on the device's MAC address. Even though this had been theorized, the result was unsatisfactory, but expected as the hardware attribute could have lowered the likelihood of a successful spoofing attack.

If the hardware attribute could have been used, the attack would have needed a step for manipulating the packets to contain the right hardware information. The fact that the hardware information needs to match would also not necessarily be known to the threat actor, adding yet another step for successfully spoofing a device. This would have, as explained in method chapter 3.4, match the thesis' predefined criteria of lowering the likelihood of an attack, as well as the criteria of cost and ease of use as registering an extra attribute is done only once.

## 5.5 Other Methods for Securing MAB

Even though the solution of using the location attribute improved MAB in certain scenarios, MAB is still not to be considered secure and other security additions in the network is needed. This section discusses some of these security additions, including Cisco Port Security and physical security. It also discusses the third research question regarding what other alternatives that exists for MAC authentication.

### 5.5.1 Port Security

The use of port security was not tested, but explored theoretically using information gathered from NTNU-IT. As theorized in chapter 2.5.2, configuring port security on active ports could potentially stop or at least temporarily halt spoofing attacks by having the port shut down when a MAC address other than the one registered tries to connect. Port security will have no effect should the malicious actor already have spoofed the device, which may be the most likely case as a malicious actor with the goal of accessing the network would come prepared.

Another problem is the scale of NTNU. The use of port security would create extra work for the employees at NTNU-IT. Configuring single ports for tens to hundreds of new devices that are to be installed across the different campuses would be time consuming and not worth the resources as it is not even guaranteed to mitigate a spoofing attack.

Port Security will based on the problems discussed above not be considered a viable solution based on the security criteria in chapter 3.4, as the added security of the solution might not be great enough to justify the cost. Had it been a viable solution to use in combination with MAB to make it more secure, the use of port security would have answered research question two (1.2.1).

### 5.5.2 Physical Security

The physical security aspects were also explored theoretically using information gathered from NTNU-IT. Physical security is an important part of securing access to wired networks, as physical access is necessary for connecting to the network. Almost all public ports available from auditoriums, classrooms, libraries, and other similar areas, are open with 802.1x as the authentication method and MAB as a fallback. This means that anyone has the possibility to connect to the wired network. Of course, as previously explained in section 3.3.3, the default VN is the "guest VN" so should one try to connect and not be able to authenticate, no network access is achieved.

To reduce the attack surface, ports not available from classrooms, labs, or auditoriums could be shut down. However, if someone has access to a building, one most often has access to classrooms as well, which would make shutting down only some ports futile.

Physical security matches the security criteria defined in chapter 3.4 as reducing the attack surface lowers the chance of an exploitation. It should therefore be considered as a part of the solution to secure MAC authentication.

### 5.5.3 Network Segmentation and Monitoring

Proper network segmentation is one solution that can justify the use of MAC authentication. The devices authorized using MAB are often technical devices and thus placed in dedicated VNs. This way, if a threat actor gets access to the network through a MAB authenticated device, they do not have access to the rest of the network. This matches the security criteria of lowering the consequences of a successful spoofing attack.

By monitoring the network closely, one can also detect a spoofing attack and neutralize the compromised device, thus limiting the impact. By having the system react automatically to an anomaly, one can limit the time the malicious device has on the network. However, this might affect other devices on the network, as false positives can lead to non-malicious devices getting blocked on the network. Procedures need to be in place for unblocking the wrongly blocked devices.

Extra work is added for both the passive and reactive solutions for blocking malicious devices, but the positive aspects of removing a malicious device's access to the network outweigh the negative aspects of the added workload. By quickly removing the malicious device's access, one reduces the impact of an exploitation as the threat actor has less time to do damage. This matches the security criteria in chapter 3.4 of lowering the consequences of a successful spoofing attack, as well as research question two.

# Chapter 6

## Conclusion

This chapter summarizes our findings and presents the conclusion of the thesis. First, each research question is repeated, and its findings are outlined. Lastly, the thesis itself is summarized before future work is presented.

### 6.1 Summary Conclusion in Relation to the Research Questions

#### 6.1.1 Research Question One

**To what extent can MAC Authentication be used to prevent unauthorized access to wired networks?**

Through both literature study and testing, the group has analyzed how MAC authentication interacts with access control principles in a wired network. The literature study showed MAC authentication is generally looked upon as insecure. However, the results from the first test where no additional attributes were used, showed that utilizing spoofing to gain access to the network could be easily done.

The solution using the custom-made location attribute together with MAC authentication could in some cases prevent unauthorized access to the network, thus making MAC authentication more secure. However, even though the location attribute was able to prevent access in some cases, MAC authentication is not to be considered a secure solution. A threat actor will still be able to access the network through MAC spoofing should they be able to match the location of the authorized MAC address.

#### 6.1.2 Research Question Two

**Are there solutions combined with MAC authentication that could make MAC authentication more secure?**

As discussed above, MAC authentication should not be considered secure. Therefore, it is important to keep this in mind and design the network to limit the consequences should someone exploit the vulnerabilities in MAC authentication. This includes segmenting the network so that similar devices and users are in their own network segments, so that if one device is spoofed, the threat actor only gets access to one part of the network and not everything. Monitoring the network will also help stop an attack early by detecting changes in the network.

Good physical security to limit access to the organization's wired network is also important, as this lowers the chance of exploitation.

Even though no solutions to make MAC authentication more secure were found, there are solutions, as mentioned above, that can be implemented in the network to make exploitation of MAC authentication less likely and less harmful.

### **6.1.3 Research Question Three**

#### **What other alternatives exist for MAC authentication?**

From the literature study, 802.1x came forward as the only good alternative for MAC authentication. It is therefore the desirable authentication method to use, but as this thesis focuses on authentication of devices not supporting 802.1, it cannot be counted as an alternative. Port Security was also explored, but deemed as not viable as an alternative, or even an addition to MAC authentication, because of the resources needed should an organization of NTNU's size implement it.

During the thesis, a viable alternative for MAC authentication for devices not supporting 802.1x was therefore not found. It can be argued that any organization utilizing devices not compatible with 802.1x should make an effort to phase out these devices. This can however prove difficult in regards to costs and logistics.

## **6.2 Thesis Summary**

The topic for this thesis was to explore secure access to wired networks, where the focus was on implementing MAC authentication in a secure manner. Implementing security features for MAC authentication was challenging due to the lack of available functionality in both ISE and in general, that we could use in conjunction with MAB. However, some possible improvements were found using custom attributes when authenticating endpoints. The possibility of adding locations was proved to prevent spoofing attacks in some cases. The location attribute was added as a custom attribute when the endpoint was authenticated using the script. This works by assigning a location to a registered MAC address as the custom attribute. If a threat actor tries to access the network using a spoofed MAC address but is connected to the wrong location, the device will not be able to access the network.

To conclude, MAC authentication can not be seen as a secure means of access control, as it is too vulnerable to spoofing. However, it is better than nothing if devices do not support 802.1x. If the problems with MAC authentication are kept in mind when designing the network architecture and security solutions to limit the consequences of an attack, this can justify the use of such an insecure authentication method.

## 6.3 Future Work

During the thesis, there were aspects that could not be explored due to time restrictions. These are deleting inactive devices from ISE and explore more attributes that can be used with MAB.

### **Delete Inactive Devices**

A desired feature that the script could have was deleting registered devices in ISE had they not been active for a given amount of time. This could, from a security perspective, help reducing the attack surface. This is due to the fact that the ones registering and mounting the devices do not necessarily delete the devices from ISE when the devices are removed and no longer in use.

### **Explore More Attributes**

Using the location attribute was a success to help mitigate a spoofing attack when MAC authentication is used, while the hardware attribute could not be used to improve the security. It would be interesting to further explore the other attributes available, like OS and device type (workstations, IoT etc.).

# Bibliography

- [1] S. Jha, *Deploying cisco ise for device administration*, eng, Oct. 2019. [Online]. Available: [https://www.cybok.org/media/downloads/Network\\_Security\\_issue\\_1.0\\_qsCh0SR.pdf](https://www.cybok.org/media/downloads/Network_Security_issue_1.0_qsCh0SR.pdf) (visited on 03/03/2023).
- [2] Cloudflare, *What is the control plane? | control plane vs. data plane*, eng. [Online]. Available: <https://www.cloudflare.com/learning/network-layer/what-is-the-control-plane/> (visited on 17/03/2023).
- [3] Cisco, *Ip routing on cisco ios, ios xe, and ios xr: How a router works*, eng, Jan. 2015. [Online]. Available: <https://www.ciscopress.com/articles/article.asp?p=2272154&seqNum=3> (visited on 27/04/2023).
- [4] Cisco, *Mac authentication bypass deployment guide*, eng, Sep. 2011. [Online]. Available: [https://www.cisco.com/c/en/us/td/docs/solutions/Enterprise/Security/TrustSec\\_1-99/MAB/MAB\\_Dep\\_Guide.html#wp392167](https://www.cisco.com/c/en/us/td/docs/solutions/Enterprise/Security/TrustSec_1-99/MAB/MAB_Dep_Guide.html#wp392167) (visited on 20/02/2023).
- [5] Cisco, *Cisco sd-access solution design guide (cvd)*, eng. [Online]. Available: <https://www.cisco.com/c/en/us/td/docs/solutions/CVD/Campus/cisco-sda-design-guide.html> (visited on 27/04/2023).
- [6] AVI Networks, *Virtual routing and forwarding (vrf)*, eng. [Online]. Available: <https://avinetworks.com/glossary/virtual-routing-and-forwarding-vrf/> (visited on 27/02/2023).
- [7] T. Keary, *A guide to spoofing attacks and how to prevent them in 2023*, eng, Mar. 2023. [Online]. Available: <https://www.comparitech.com/net-admin/spoofing-attacks-guide/> (visited on 10/05/2023).
- [8] M. P. Teng Xu James B. Wendt, *Security of iot systems: Design challenges and opportunities*, eng, Nov. 2014. [Online]. Available: <https://ieeexplore.ieee.org/abstract/document/7001385> (visited on 20/04/2023).
- [9] K. S. Mousa Alramadhan, *An overview of access control mechanisms for internet of things*, 2017. DOI: 10.1109/ICCCN.2017.8038503.
- [10] F. Schneider, 'Least privilege and more [computer security],' *IEEE Security & Privacy*, vol. 1, no. 5, pp. 55–59, 2003. DOI: 10.1109/MSECP.2003.1236236.
- [11] S. Rose, O. Borchert, S. Mitchell and S. Connelly, 'Zero Trust Architecture,' en, National Institute of Standards and Technology, Tech. Rep., Aug. 2020. DOI: 10.6028/NIST.SP.800-207. [Online]. Available: <https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-207.pdf> (visited on 16/02/2023).



- [12] C. Buck, C. Olenberger, A. Schweizer, F. Völter and T. Eymann, 'Never trust, always verify: A multivocal literature review on current knowledge and research gaps of zero-trust,' en, *Computers & Security*, vol. 110, p. 102 436, Nov. 2021, ISSN: 0167-4048. DOI: 10.1016/j.cose.2021.102436. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0167404821002601> (visited on 16/02/2023).
- [13] A. D. Gupta, B. G. Tiwari, C. Y. Kapoor and D. P. Kumar, *Media access control (mac) mac spoofing and its countermeasures*, eng, Nov. 2009. [Online]. Available: <https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=a033e7d325a70917415b8e55c17683fb82> (visited on 13/02/2023).
- [14] E. D. Cardenas, *Mac spoofing—an introduction*, eng, Aug. 2003. [Online]. Available: <https://www.giac.org/paper/gsec/3199/mac-spoofing-an-introduction/105315> (visited on 16/02/2023).
- [15] Cisco, *Osi model chart*, eng. [Online]. Available: <https://learningnetwork.cisco.com/s/article/osi-model-reference-chart> (visited on 25/02/2023).
- [16] Carnegie Mellon University, *Mac address (media access control address)*, eng. [Online]. Available: <https://www.cmu.edu/computing/services/endpoint/network-access/mac-address.html> (visited on 09/02/2023).
- [17] Cloud RADIUS, *Radius servers for noobs: Everything you need to know*, eng. [Online]. Available: <https://www.cloudradius.com/a-complete-guide-to-radius-servers/> (visited on 14/02/2023).
- [18] C. Rigney, S. W. Livingston and W. S. A. Rubens Merit, *Remote authentication dial in user service (radius)*, eng, Jun. 2000. [Online]. Available: <https://www.rfc-editor.org/rfc/rfc2865> (visited on 18/05/2023).
- [19] Cisco, *Mac authentication bypass*, eng. [Online]. Available: [https://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst2960xr/software/15-2\\_5\\_e/configuration\\_guide/b\\_1525e\\_consolidated\\_2960xr\\_cg/mac\\_authentication\\_bypass.pdf](https://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst2960xr/software/15-2_5_e/configuration_guide/b_1525e_consolidated_2960xr_cg/mac_authentication_bypass.pdf) (visited on 31/01/2023).
- [20] Cisco, *Understanding radius*, eng, Jun. 2007. [Online]. Available: [https://www.cisco.com/c/en/us/td/docs/net\\_mgmt/access\\_registrar/1-7/concepts/guide/radius.html](https://www.cisco.com/c/en/us/td/docs/net_mgmt/access_registrar/1-7/concepts/guide/radius.html) (visited on 23/01/2023).
- [21] Huawei, *How does radius work?* eng, Sep. 2022. [Online]. Available: <https://support.huawei.com/enterprise/en/doc/EDOC1100086516> (visited on 17/02/2023).
- [22] T. Kumpulainen, *Network access control*, eng, Sep. 2020. [Online]. Available: <https://www.theseus.fi/bitstream/handle/10024/345970/Kumpulainen%5C%20Topias.pdf?sequence=2&isAllowed=y> (visited on 23/01/2023).
- [23] J. Hill, *An analysis of the radius authentication protocol*, eng, Nov. 2001. [Online]. Available: <http://lms.uni-mb.si/~meolic/ptk-seminarske/radius.pdf> (visited on 23/01/2023).
- [24] Deepesh Sharma, *How to change your mac address on linux*, eng, Sep. 2021. [Online]. Available: <https://www.makeuseof.com/how-to-change-mac-address-on-linux/> (visited on 16/02/2023).
- [25] J. Feng, *Analysis, implementation and extensions of radius protocol*, eng, Jun. 2009. [Online]. Available: <https://ieeexplore.ieee.org/abstract/document/5116234> (visited on 02/03/2023).

- [26] Cisco, *Wired 802.1x deployment guide*, eng, Sep. 2011. [Online]. Available: [https://www.cisco.com/c/en/us/td/docs/solutions/Enterprise/Security/TrustSec\\_1-99/Dot1X\\_Deployment/Dot1x\\_Dep\\_Guide.html#wp386716](https://www.cisco.com/c/en/us/td/docs/solutions/Enterprise/Security/TrustSec_1-99/Dot1X_Deployment/Dot1x_Dep_Guide.html#wp386716) (visited on 13/02/2023).
- [27] D. Zeni, *Cisco ise: Wired and wireless 802.1x network authentication*, eng. [Online]. Available: <https://www.lookingpoint.com/blog/ise-series-802.1x> (visited on 17/02/2023).
- [28] E. L. Brown, *802.1X Port-Based Authentication*, 1st ed. Auerbach Publications, 2006, pp. 1–37.
- [29] Cisco, *Catalyst 4500 series switch cisco ios software configuration guide*, eng, May 2022. [Online]. Available: <https://www.cisco.com/c/dam/en/us/td/docs/switches/lan/catalyst4500/XE3-11-0E/configuration/guide/xs-311-cg.pdf> (visited on 13/02/2023).
- [30] U. Khan, *Implementing scalable security for devices without 802.1x support*, en-us, Nov. 2022. [Online]. Available: <https://www.sans.org/white-papers/implementing-scalable-security-for-devices-without-802-1x-support/> (visited on 05/01/2023).
- [31] Cisco, *Configuring port security*, eng, May 2007. [Online]. Available: [https://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst4000/8-2glx/configuration/guide/sec\\_port.html#wp1019841](https://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst4000/8-2glx/configuration/guide/sec_port.html#wp1019841) (visited on 14/02/2023).
- [32] H. S. K. Sihyung Lee Kyriaki Levanti, *Network monitoring: Present and future*, en-us, Feb. 2013. [Online]. Available: [https://www.sciencedirect.com/science/article/pii/S138912861400111X?casa\\_token=jIwx03i2yIcAAAAA:K8ogtJc1w-h0a8c6G01s95mb1SNZvmFwkfzZZjRbqDp](https://www.sciencedirect.com/science/article/pii/S138912861400111X?casa_token=jIwx03i2yIcAAAAA:K8ogtJc1w-h0a8c6G01s95mb1SNZvmFwkfzZZjRbqDp) (visited on 05/01/2023).
- [33] Y. S. Ehsan Saboori Shafiq Parsazad, *Automatic firewall rules generator for anomaly detection systems with apriori algorithm*, eng, Sep. 2010. [Online]. Available: <https://ieeexplore.ieee.org/document/5579365> (visited on 15/05/2023).
- [34] R. Chandramouli, *Guide to a secure enterprise network landscape*, en-us, Nov. 2022. [Online]. Available: <https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-215.pdf> (visited on 05/02/2023).
- [35] N. F. Hyojoon Kim, *Improving network management with software defined networking*, eng, Feb. 2013. [Online]. Available: <https://ieeexplore.ieee.org/abstract/document/6461195> (visited on 20/03/2023).
- [36] S. V. Jason Gooley Roddie Hasan, *Cisco Software-Defined Access*, 1st ed. Hoboken, NJ: Cisco Press, 2021.
- [37] D. Zeni, *Cisco SD-Access*, en-us. [Online]. Available: <https://www.lookingpoint.com/blog/sd-access> (visited on 23/02/2023).
- [38] Cisco, *Cisco identity services engine administrator guide, release 3.2*, eng. [Online]. Available: [https://www.cisco.com/c/en/us/td/docs/security/ise/3-2/admin\\_guide/b\\_ise\\_admin\\_3\\_2/b\\_ISE\\_admin\\_32\\_overview.html](https://www.cisco.com/c/en/us/td/docs/security/ise/3-2/admin_guide/b_ise_admin_3_2/b_ISE_admin_32_overview.html) (visited on 19/04/2023).
- [39] K. Thiruvengadam, *Network security knowledge area*, eng, Feb. 2018. [Online]. Available: <https://community.cisco.com/t5/security-knowledge-base/cisco-ise-device-administration-prescriptive-deployment-guide/ta-p/3738365#toc-hId-1206426492> (visited on 08/03/2023).

- [40] Cisco, *Cisco identity services engine administrator guide, release 2.0*, eng, Oct. 2019. [Online]. Available: [https://www.cisco.com/c/en/us/td/docs/security/ise/2-0/admin\\_guide/b\\_ise\\_admin\\_guide\\_20/b\\_ise\\_admin\\_guide\\_20\\_chapter\\_010010.html](https://www.cisco.com/c/en/us/td/docs/security/ise/2-0/admin_guide/b_ise_admin_guide_20/b_ise_admin_guide_20_chapter_010010.html) (visited on 16/03/2023).
- [41] *HTTP request methods - HTTP | MDN*, en-US, Apr. 2023. [Online]. Available: <https://developer.mozilla.org/en-US/docs/Web/HTTP/Methods> (visited on 18/05/2023).
- [42] D. Sutton, *Information Risk Management*, 2nd ed. BCS THE CHARTERED INSTITUTE FOR IT, 2021, pp. 1–219.
- [43] S. Venaas, K. Wierenga, S. Winter and M. McCauley, *Transport layer security (tls) encryption for radius*, eng, May 2012. [Online]. Available: <https://datatracker.ietf.org/doc/html/rfc6614> (visited on 16/05/2023).
- [44] Cisco, *Managing network devices*, eng. [Online]. Available: [https://www.cisco.com/en/US/docs/security/ise/1.0/user\\_guide/ise10\\_man\\_network\\_devices.html#wp1115233](https://www.cisco.com/en/US/docs/security/ise/1.0/user_guide/ise10_man_network_devices.html#wp1115233) (visited on 20/04/2023).

## **Appendix A**

# **Python Code**

## A.1 Script: Main Interface

```

from endpointgroup import *
from editEndpoint import *
from addEndpoint import *
from addBulk import *

#Interface funksjon
def main():
    print("Cisco_ISE_Endpoint_manager")
    #Input valg
    choice = input("Press_1_to_list_all_endpoints_in_group,_press_2_to_display_single_endpoint_
        by_MAC,_press_3_to_create_endpoint,_press_4_to_update_endpoint,_press_5_to_delete_
        endpoint,_press_6_to_add_endpoints_in_bulk,_press_any_other_key_to_exit\n")
    if choice == "1": #Valg for å liste alle endepunkt
        print("List_all_endpoints_by_group\n")
        groupname = input("Please_enter_name_of_group\n")
        listendpointbygroup(groupname)
    elif choice == "2": #Valg for å liste spesefikt endepunkt
        print("List_single_endpoint_by_MAC-address\n")
        mac = input("Enter_MAC-address:\n")
        listendpointbymac(mac)
    elif choice == "3": #Valg for å lage endepunkt
        print("Create_endpoint")
        add_Endpoint()
    elif choice == "4": #Valg for å oppdatere endepunkt
        print("Edit_endpoint")
        updateEndpoint()
    elif choice == "5": #Valg for å slette endepunkt
        print("Delete_endpoint\n")
        macaddress = input("Enter_endpoint_for_deletion\n")
        print("Are_you_sure_you_want_to_delete_endpoint_{}?".format(macaddress))
        deletechoice = input("Y/N")#Ja/Nei
        if delchoice == "Y" or delchoice == "y": #Bekreftede valg
            deleteEndpoint(macaddress)
        else:
            exit()
    elif choice == "6": #Valg for å lage flere endepunkt samtidig
        print("Add_endpoints_in_bulk\n")
        filepath = input("Enter_filepath\n")
        add_Bulk(filepath)
    else: #Avslutter programmet
        print("Exiting_program")
        exit()

main()

```

## A.2 Script: Add Endpoint

```

import json
import requests
import warnings
import logging as log
import sys
warnings.filterwarnings("ignore")
from requests.auth import HTTPBasicAuth
from endpointgroup import listgroup
from macvalidator import validator

user = ""#Brukernavn
pwd = ""#Passord
base_url = "https://sand-iseadm01.nettel.ntnu.no:443/"
basic_auth = HTTPBasicAuth(user, pwd)

#Funksjon for å liste alle endepunkter
def list_endpoints():
    response = requests.get(url=base_url+"api/v1/endpoint",auth=basic_auth,verify=False)
    print(response.text)

#Funksjon for å legge til endepunkt
def add_Endpoint():
    mac = input("Please_enter_mac_address\n")

    #Validering av MAC-adresse formatering
    if (validator(mac) == False):
        print("Invalid_MAC-Address")
        quit()
    #Input felt
    groupname = input("Please_enter_the_name_of_the_group_for_the_endpoint\n")
    groupId = listgroup(groupname)
    endpointname = input("Please_enter_name_of_endpoint\n")
    endpointdesc = input("Please_enter_description_of_endpoint\n")
    location = input("Please_enter_location_of_input\n")

    payload = {
        "groupId": "{}".format(groupId), #*
        "description": "{}".format(endpointdesc),
        "mac": "{}".format(mac),#*
        "name": "{}".format(endpointname),#*
        "staticGroupAssignment": True,#*
        "staticProfileAssignment": True,#*
        "customAttributes": {
            "fms_location": location,
        }
    }
    headers = {
        "content-type": "application/json",
        "accept": "application/json",
        "cache-control": "no-cache",
        "Connection": "close",
    }
    response_new_endpoint = requests.post(url=base_url+"api/v1/endpoint",auth=basic_auth,verify=False,headers=headers,data=json.dumps(payload))

    print(response.request.body) #Forespørsel innhold
    print(response_new_endpoint.json()) #Respons innhold

```

### A.3 Script: Edit and Delete Endpoint

```

import json
import requests
import warnings
import logging as log
import sys
from requests.auth import HTTPBasicAuth
warnings.filterwarnings("ignore")
from endpointgroup import listgroup
from macvalidator import validator

user = ""#Brukernavn
pwd = ""#Passord

#Funksjon for å liste alle endepunkter
def listallendpoints():

    headers = {
        "Connection": "close",
        'content-type': "application/json",
        'accept': "application/json",
        'cache-control': "no-cache",
        'Connection': "close",
    }

    response = requests.get('https://sand-iseadm01.nettel.ntnu.no:9060/ers/config/endpoint?size
=100', auth=(user, pwd), verify=False, headers=headers)
    print(response)
    print(response.text)

#Funksjon for å liste alle endepunkter i en gruppe
def listendpointbygroup(groupname):

    groupId = listgroup(groupname)
    headers = {
        "Connection": "close",
        'content-type': "application/json",
        'accept': "application/json",
        'cache-control': "no-cache",
        'Connection': "close",
    }

    response = requests.get('https://sand-iseadm01.nettel.ntnu.no:9060/ers/config/endpoint?
filter=groupId.eq.{}'.format(groupId), auth=(user, pwd), verify=False, headers=headers)
    print(response) #Responskode
    print(response.text) #Respons tekst

#Funksjon for å lise endepunkt etter MAC-adresse
def listendpointbymac(mac):

    headers = {
        "Connection": "close",
        'content-type': "application/json",
        'accept': "application/json",
        'cache-control': "no-cache",
        'Connection': "close",
    }

    response = requests.get('https://sand-iseadm01.nettel.ntnu.no:9060/ers/config/endpoint/name
/{}'.format(mac), auth=(user, pwd), verify=False, headers=headers)

    print(response) #Responskode
    print(response.text) #Respons tekst
    output = (json.loads(response.content)) #Henter ut responsinnhold i JSON-format
    return output["ERSEndPoint"]["id"] #Returnerer endepunkts-ID for bruk i andre funksjoner

#Funksjon for å oppdatere endpoint

```

```

def updateEndpoint():
    endpointname = input("Enter_endpoint_name:\n") #Input gammel navn
    endpointId = listendpointbymac(endpointname)
    newgroup = input("Enter_group_name_for_endpoint\n") #Input gruppetilhørighet
    newgroupId = listgroup(newgroup)
    print(newgroupId)
    newdesc = input("Enter_new_endpoint_description:\n") #Input ny beskrivelse
    payload = {
        "ERSEndPoint" : {
            "groupId": newgroupId,
            "description": newdesc,
            "mac": endpointname,
            "name": endpointname,
            "staticGroupAssignment": True,
            "staticProfileAssignment": True,
            "customAttributes": {
                "customAttributes": {
                    "fms_location": location
                }
            }
        }
    }
    headers = {
        'content-type': "application/json",
        'accept': "application/json",
        'cache-control': "no-cache",
        'Connection': "close",
    }
    print(headers)
    response = requests.put('https://sand-iseadm01.nettel.ntnu.no:9060/ers/config/endpoint/{}'.format(endpointId),
        auth=(user, pwd),
        verify=False,
        headers=headers,
        data=json.dumps(payload),
    )
    print(response) #Responskode
    print(response.text) #Responstekst
    print(response.request.body) #Responsinnhold

#Funksjon for å slette endepunkt
def deleteEndpoint(macaddress):

    print("Are you sure you want to delete MAC-address: {}".format(macaddress))

    macId = listendpointbymac(macaddress)
    headers = {
        'content-type': "application/json",
        'accept': "application/json",
        'cache-control': "no-cache",
        'Connection': "close",
    }
    response = requests.delete('https://sand-iseadm01.nettel.ntnu.no:9060/ers/config/endpoint/{}'.format(macId),
        auth=(user, pwd),
        verify=False,
        headers=headers,
    )
    print(response)#Responskode

```



## A.4 Script: Add Endpoints in Bulk

```
import json
import requests
import warnings
import logging as log
import sys
warnings.filterwarnings("ignore")
from requests.auth import HTTPBasicAuth
from endpointgroup import listgroup
from macvalidator import validator
from editEndpoint import listendpointbymax

user = "#Brukernavn"
pwd = "#Passord"

def add_Bulk(path): #Funksjon for å legge til flere endepunkt samtidig fra en JSON fil

    with open(path, 'r') as f: #Avlesning fra JSON fil
        contents = json.loads(f.read())

    payload = contents["endpoints"]
    headers = {
        "content-type": "application/json",
        "accept": "application/json",
        "cache-control": "no-cache",
        "Connection": "close",
    }
    response = requests.post(url=base_url+"api/v1/endpoint/bulk", auth=basic_auth, verify=False,
        headers=headers, data=json.dumps(payload))
    print(response) #Respons kode
    print(response.request.body) #Respons innhold
    print(response.json())
```

## A.5 Script: MAC-address Validator

```
import re
def validator(macaddr): #Funksjon for å validere MAC-adresse formatering

    #Godkjente formater
    #AA:BB:CC:DD:EE:FF
    #aa:bb:cc:dd:ee:ff
    #AA-BB-CC-DD-EE-FF
    #aa-bb-cc-dd-ee-ff
    #AAAA.BBBB.CCCC
    #aaaa.bbbb.cccc
    #AAAAAAAAAAAA
    #aaaaaaaaaaaa

    pattern = ("^([0-9A-Fa-f]{2}[:-])" +
               "{5}([0-9A-Fa-f]{2})|" +
               "([0-9a-fA-F]{4}\\." +
               "[0-9a-fA-F]{4}\\." +
               "[0-9a-fA-F]{4})|" +
               "[0-9a-fA-F]{12}$")

    regex = re.compile(pattern)

    if (macaddr == None):
        return False

    if(re.search(regex, macaddr)): #Returnerer True hvis godkjent
        return True
    else:
        return False
```

## A.6 Example JSON file

```
{
  "endpoints":[
    {
      "groupId": "afb89b40-bcbf-11ed-b7b0-728cbce82793",
      "description": "Bulk endpoint test 1",
      "mac": "CC:CC:CC:CC:CC:CC",
      "name": "Bulk1",
      "staticGroupAssignment": true,
      "staticProfileAssignment": true,
      "customAttributes": {
        "fms_location": "Sluppen"
      }
    },
    {
      "groupId": "afb89b40-bcbf-11ed-b7b0-728cbce82793",
      "description": "Bulk endpoint test 2",
      "mac": "DD:DD:DD:DD:DD:DD",
      "name": "Bulk2",
      "staticGroupAssignment": true,
      "staticProfileAssignment": true,
      "customAttributes": {
        "fms_location": "Sluppen"
      }
    }
  ]
}
```

**Figure A.1:** A template for adding JSON files in bulk



 **NTNU**

Norwegian University of  
Science and Technology