Final Engagement

Defense of a Vulnerable Network

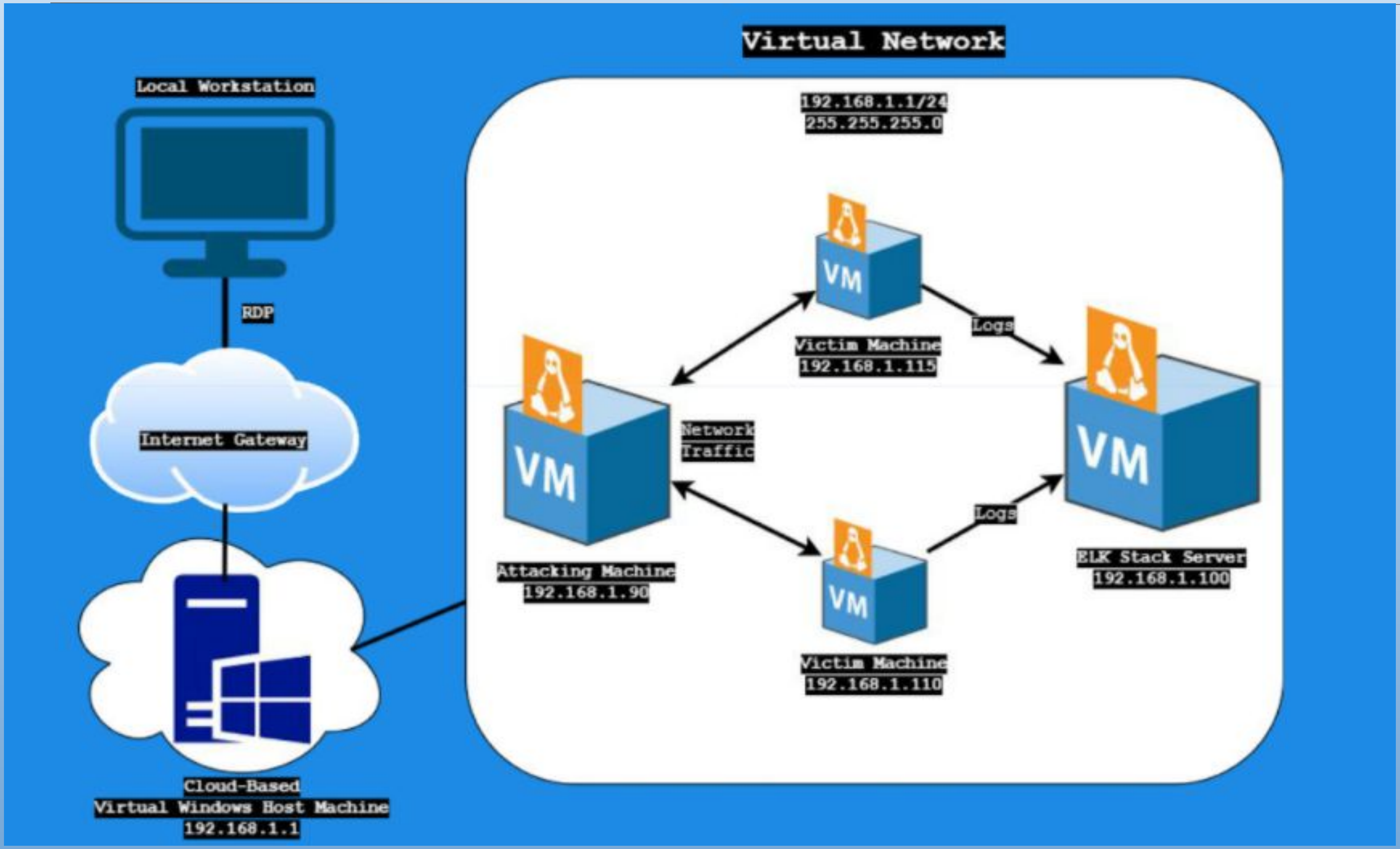
Conner Gadwood, Damien Lee, Emmanuel Oniovosa and Tanner White

Table of Contents

This document contains the following resources:

Network Topology
& Critical Vulnerabilities

Network Topology



Network
Address
Range:192.168.1.0/24
Netmask:255.255.255.0
Gateway: 192.168.1.1

Machines
IPv4:192.168.1.90
OS: Kali Linux
Hostname: Kali

IPv4:192.168.1.110
OS: Linux
Hostname: Target 1

IPv4: 192.168.1.115
OS: Linux
Hostname: Target 2

IPv4: 192.168.1.100
OS: Linux
Hostname: Elk

Critical Vulnerabilities: Target 1

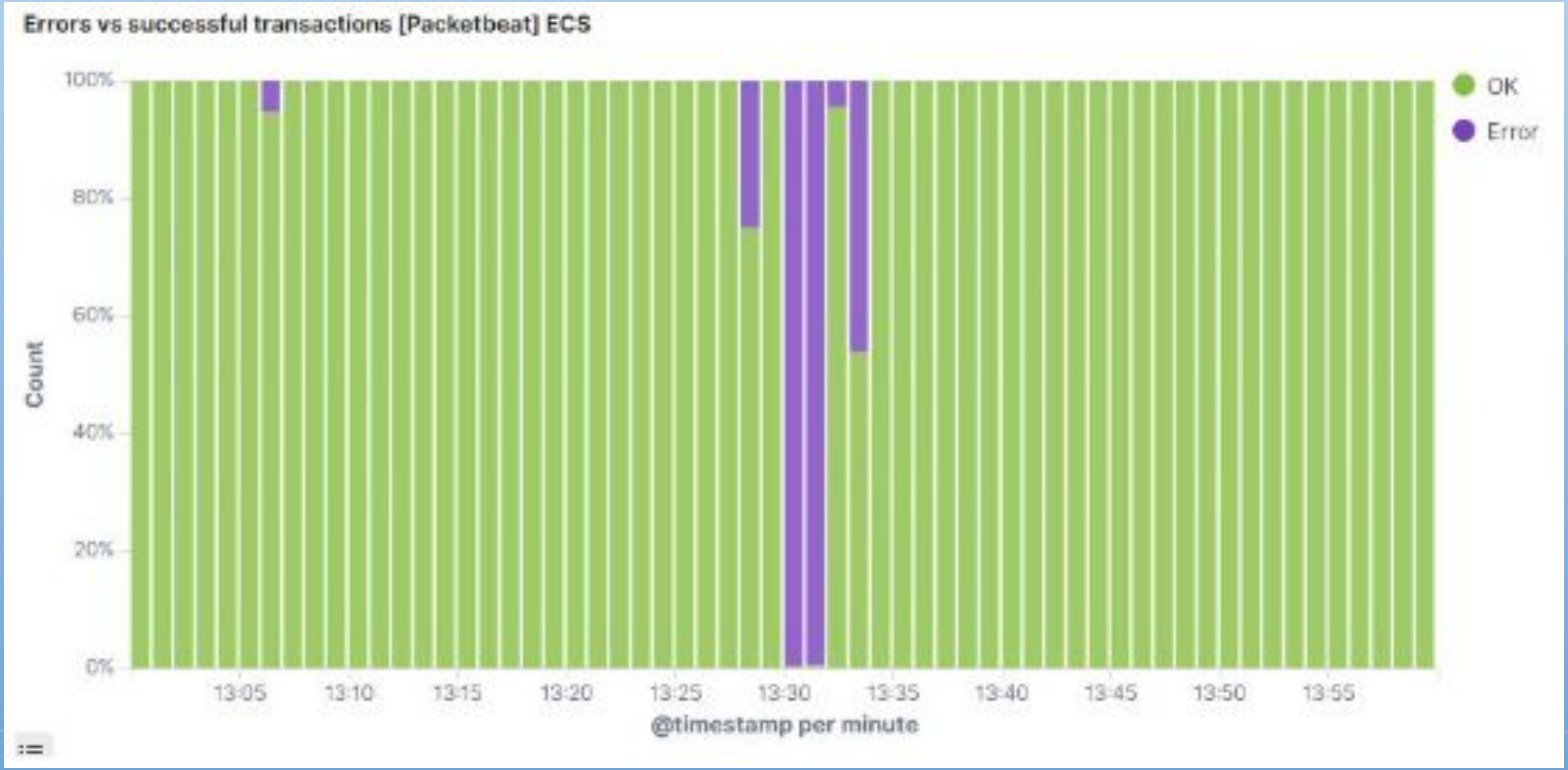
Our assessment uncovered the following critical vulnerabilities in **Target 1**.

Vulnerability	Description	Impact
Weak Passwords	Passwords did not require complex characters/symbols nor did they require set time resets	Passwords can easily brute-forced
Admin login credentials stored on the public server file	The file wp-config.php contains the admin login and password	This could allow anyone root access
Accessible user password hashes	Password hashes are stored on the MySQL database	A malicious actor can obtain root access wit this method
Python root escalation	The user Steven can use python scripting to spawn a bash shell and escalate to root privileges	An actor can obtain root access with this method

Alerts Implemented

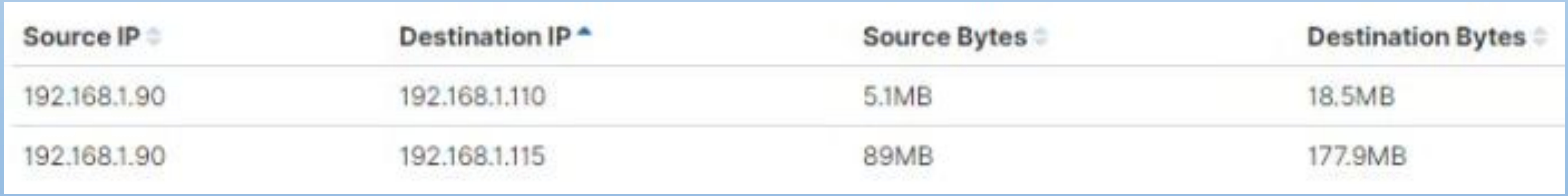
Excessive HTTP Errors

- This alert keeps track of any incoming HTTP error codes
- It is set to fire if more than 400 error codes are detected in the span of 5 minutes.



HTTP Request Size Monitor

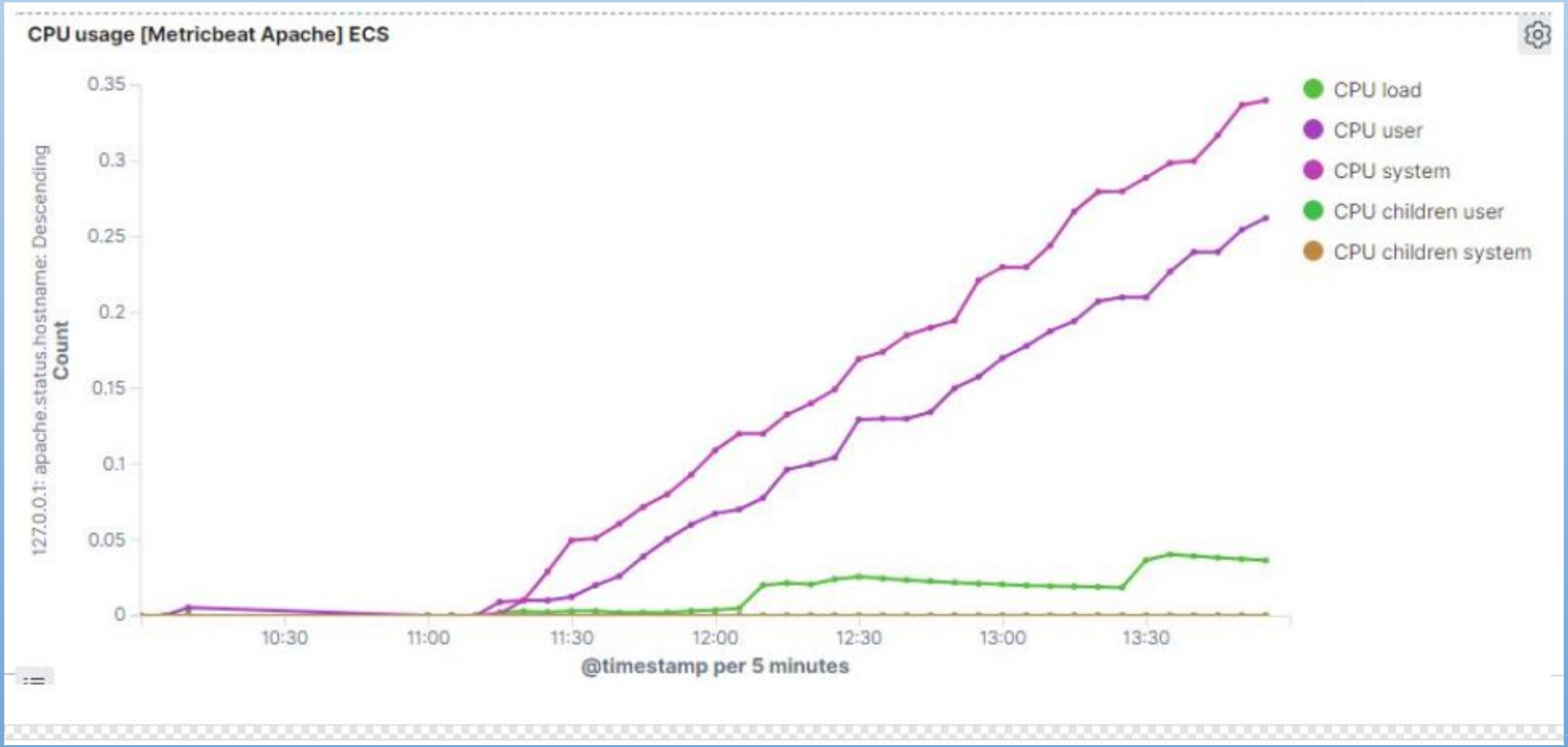
- This alert monitors the size of all incoming HTTP requests
- It is set up to fire upon receiving any HTTP request that is over 3500 bytes in size



Source IP	Destination IP	Source Bytes	Destination Bytes
192.168.1.90	192.168.1.110	5.1MB	18.5MB
192.168.1.90	192.168.1.115	89MB	177.9MB

CPU Usage Monitor

- Monitors percentage of total CPU usage since last event.
- Threshold fires if 50% is used within the last 5 minutes.



Hardening

Weak Passwords

- Creating a stronger password policy to include complex characters could be an easy solution to combat brute force attacks.
  - A stronger password policy would be implemented by the department in charge of the password and usernames
  - Adding things such as:
    - Password history
    - Password length
    - Complex characters
    - Minimum and/or maximum age

Prevent User Enumeration

WordPress leaks usernames in several ways. In truth there is no way to fully prevent user enumeration, particularly if your website makes use of authors pages. However, you can certainly reduce the attack surface and make user enumeration harder by following the below steps

- ❑ **Disable the WordPress REST API if you are not using it,**
- ❑ **Disable WordPress XML-RPC if you are not using it,**
- ❑ **Configure your web server to block requests to /?author=<number>,**
- ❑ **Don’t expose /wp-admin and /wp-login.php directly to the public Internet.**

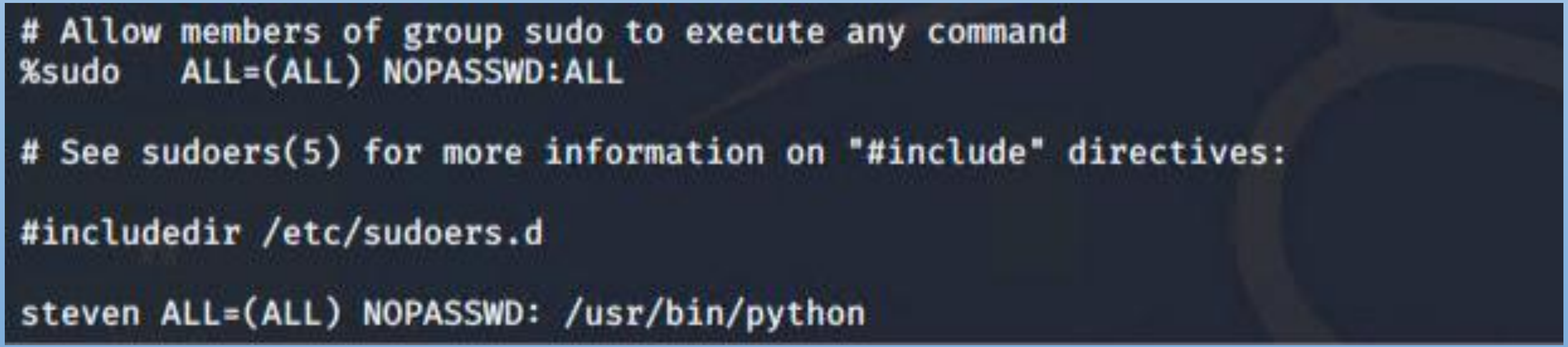
Password Protected Directories

- Creating directories that are password protected that contain the sensitive and important information could be a good way to make sure the general users cannot access this information
- There are many great applications to use in order to protect these directories such as
    - CyberSafe- top secret information program that uses modern encryption algorithms
    - USB Safeguard-software to encrypt and protect data with a password on a removable pen drive (must have removable pen drives in use for this)
    - Proxycrypt-command line tool that creates encrypted volumes within a file or hard drive

Implementing Patches

Restricting python Root Access

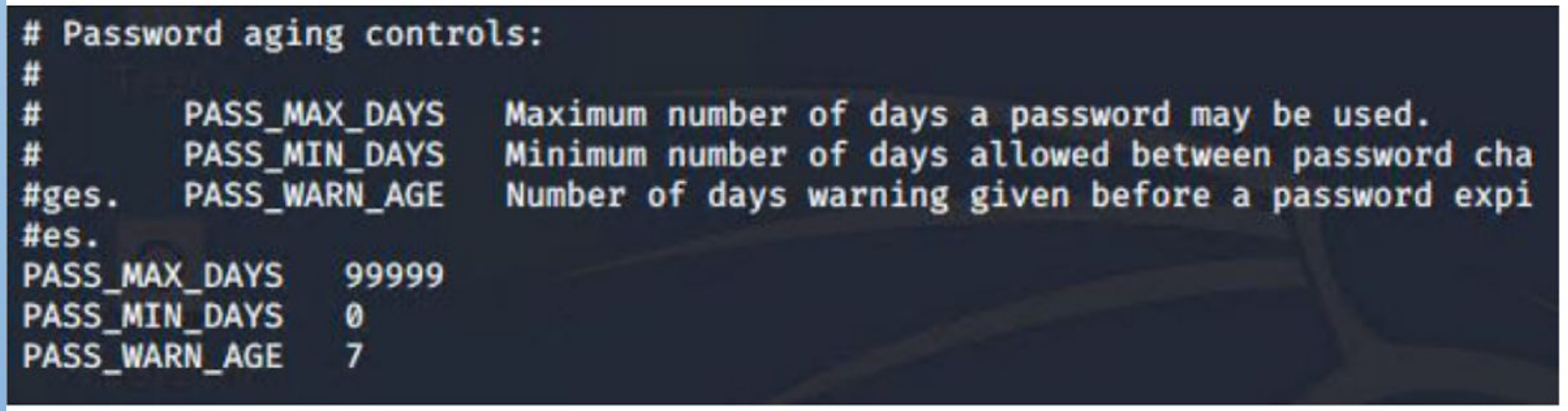
- Restricting Steven's Python abilities to that of an average user closes this vulnerability.
- To remove Steven's privileges, all we have to do is remove him from the sudoers file.
- This works because it will no longer give Steven Root abilities.



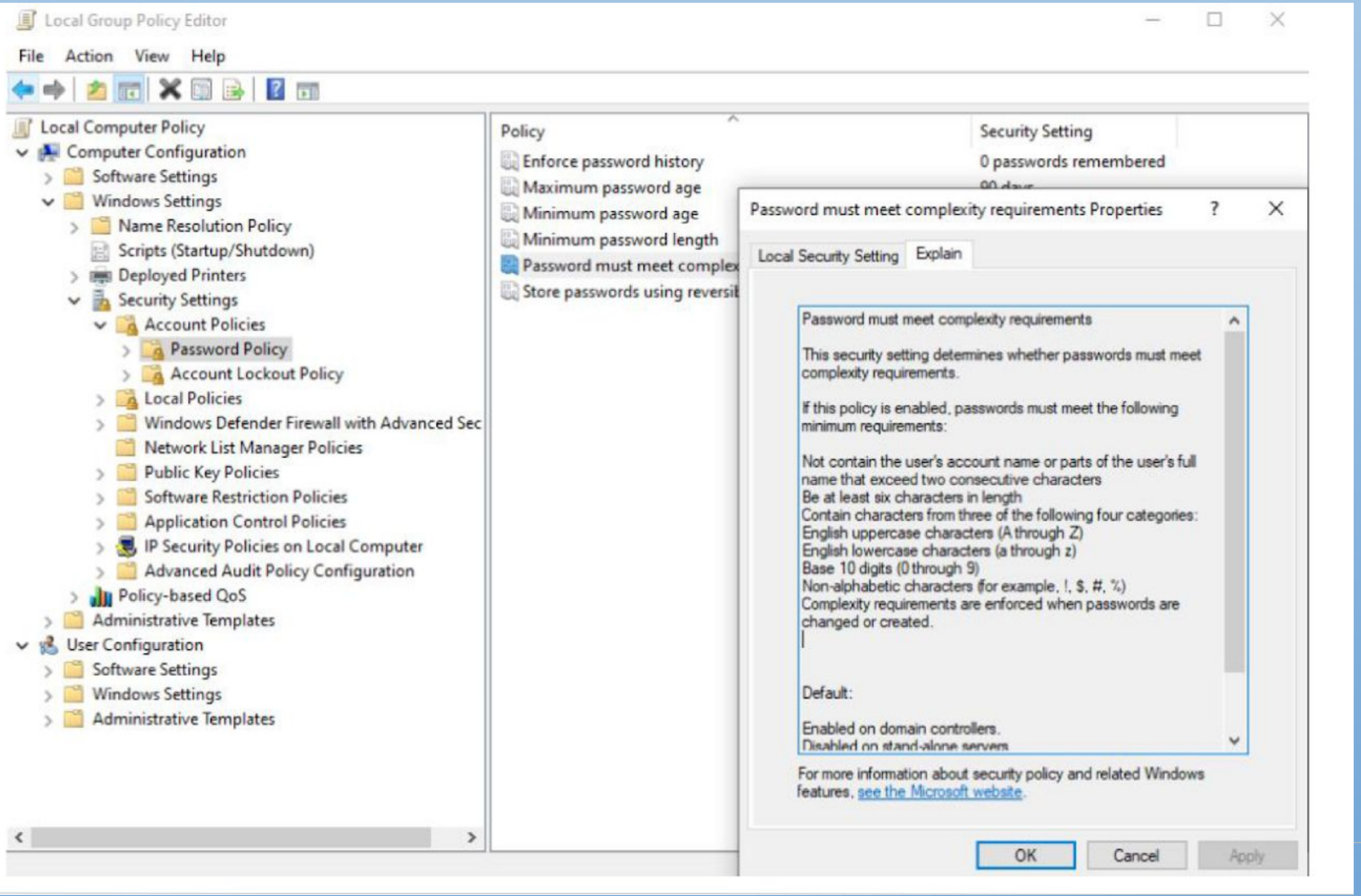
```
# Allow members of group sudo to execute any command
%sudo   ALL=(ALL) NOPASSWD:ALL

# See sudoers(5) for more information on "#include" directives:

#includedir /etc/sudoers.d

steven ALL=(ALL) NOPASSWD: /usr/bin/python
```

Creating a Password Policy

- Depending on what environment we are working in, there are a couple of different ways we can manage password policies.
- If we are working in Linux we can simply edit the login.defs file, located in the /etc directory.
- As for Windows, we have full control over user password policies using the Local Group Policy Editor.



```
# Password aging controls:
#
#       PASS_MAX_DAYS   Maximum number of days a password may be used.
#       PASS_MIN_DAYS   Minimum number of days allowed between password cha
#ges.   PASS_WARN_AGE   Number of days warning given before a password expi
#es.
PASS_MAX_DAYS   99999
PASS_MIN_DAYS   0
PASS_WARN_AGE   7
```



Policy	Security Setting
Enforce password history	0 passwords remembered
Maximum password age	90 days
Minimum password age	
Minimum password length	
Password must meet complex	
Store passwords using reversi	

Password must meet complexity requirements Properties

Password must meet complexity requirements

This security setting determines whether passwords must meet complexity requirements.

If this policy is enabled, passwords must meet the following minimum requirements:

Not contain the user's account name or parts of the user's full name that exceed two consecutive characters
Be at least six characters in length
Contain characters from three of the following four categories:
English uppercase characters (A through Z)
English lowercase characters (a through z)
Base 10 digits (0 through 9)
Non-alphabetic characters (for example, !, $, #, %)
Complexity requirements are enforced when passwords are changed or created.

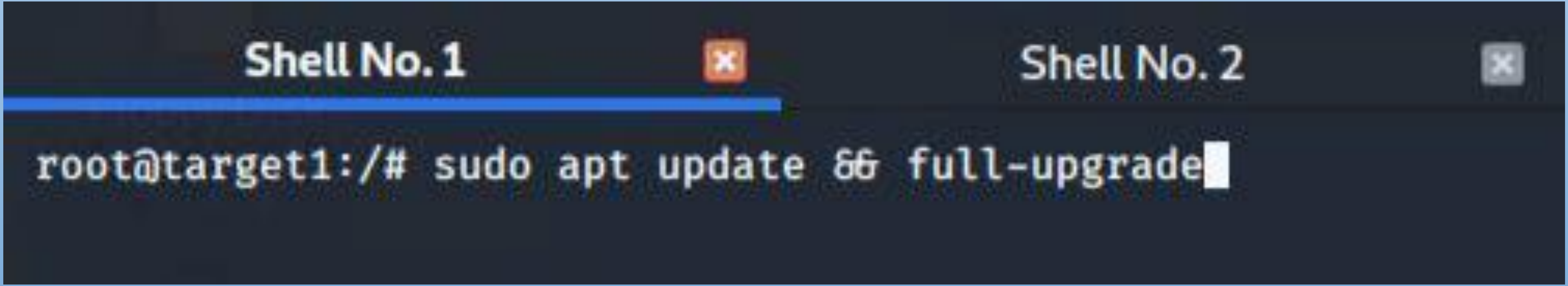
Default:

Enabled on domain controllers.
Disabled on stand-alone servers

For more information about security policy and related Windows features, see the Microsoft website.

Update the Operating System

- Last but not least. We need to make sure that our systems are up to date. This is more of a general security tip than a mitigation of any specific vulnerability.
- The best way to be safe from known vulnerabilities is to install the latest security update for your operating system with the following command line script.
- Going forward you should constantly be doing updates on a regular time frame.



```
root@target1:/# sudo apt update && full-upgrade
```