JOHNS HOPKINS
WHITING SCHOOL
of ENGINEERING
Information Security Institute

EN.650.624, Network Security, Spring 2020

Instructor: Reuben Johnston, reub@jhu.edu

Homework 5

See Blackboard for grade %

Due Date

See Blackboard for date

Summary

In this homework exercise, students will setup and use a Kerberos server for authenticating an ad-hoc network between systems they control on our ns-public network. Additionally, they will use SIEM tools to monitor elements on the network and perform attacks on other student networks that are identified.

Part 1

- Setup an insecure NFS server (using rw, insecure, and no_root_squash attributes) in Docker container on raspberry pi (RPI) or VM (running Ubuntu server); create several users with easy to guess usernames/passwords and install an openssh-server; this server needs to be running on our ns-public network so that other student teams can attack it
- Place a goldenkey.txt file somewhere on the server (with your team name in it) in a folder that is not accessible from the NFS share and set the folder access to require root privileges
- Setup Kali virtual machine (VM) on same network with server and prepare Wireshark to capture a network sequence

Attacks

1. Mount the NFS file share from another RPI or VM and capture the network sequence when accessing a file remotely
2. Setup an email server as we did in HW4 in a Docker container on the same attacker VM; create an email account named nwsec on the email server
3. Use hydra to perform ssh username/password guessing attacks on another team's NFS server from another RPI or VM
4. Abuse the NFS share by uploading a malicious executable that sets the current user to root when it is executed
5. Login to the server using your credentials obtained via hydra and run your malicious executable to obtain root privileges; locate and download the goldenkey.txt file

Part 2

- Setup a Kerberos server and a second secure NFS server that requires Kerberos authentication for access to it
- Create several network authenticate users on the system and install an openssh-server that uses Kerberos authentication
- Repeat the step one from part 1 and capture the network sequence (this time authenticating with Kerberos)

Part 3

- Install Suricata in another Docker container or VM and configure it to detect attacks on the insecure NFS server (you will need to mirror the ports so that Suricata can see the traffic)
- Identify any attackers ip addresses who are accessing your server during the CTF and send them a funny email to address nwsec@ipaddressofattacker that asks them to please stop being a bad neighbor

Part 4

- Answer the questions

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING | Information Security Institute

EN.650.624, Network Security, Spring 2020 Instructor: Reuben Johnston, reub@jhu.edu

Requirements

Executive summary report

The report must contain an overview of the project, the project goals, project execution highlights (i.e., what was performed, with pertinent technical highlights), what goals were accomplished (and any that were not achieved), and what was learned from the project (2-3 pages). For the report format, groups shall use the IEEE format (see here). The report is intended to be reference material for when you are performing threat modeling in practice.

Groups will begin the project with the creation of a project backlog and by assigning tasks in the backlog amongst the team. For each task in the backlog, groups should include exit conditions and schedule milestones.

For this assignment, place supplementary code and documentation in the specified locations on the course GitHub. Submit supplemental *.pcap files in a single *.zip archive using BB. Turn in the executive summary report, via TurnItIn assignment on BB, and your supplemental *.zip in the separate BB assignment for supplemental material. Please include team member last names in the filenames for all files submitted (e.g., "hw5-execsum-name1-name2-name3.docx" or hw5-supplemental-name1-name2-name3.zip).

Deliverables

Part 1

1. Describe the sequence used
2. Provide a python script that executes your attack
3. Provide *.pcap network capture file from traffic between attack host and target running the application server
4. Describe the results of the attack

Part 2

1. Provide the details for installing Kerberos server, authenticated network users, and NFS/ssh servers authenticating using Kerberos
2. Provide *.pcap network capture file from traffic between authenticated client and application server
3. Describe the advantages to this setup vs that in part 1

Part 3

1. Provide the details used to install and configure Suricata so that it could be replicated by someone else
2. Provide reporting details from Suricata that show it detected the attack

Part 4

1. Discuss what else could be performed to defend against these attacks
2. Initial versions of NFS were not a secure protocol. Discuss some of the secure protocol alternatives available today (i.e., related to privacy).

Grading Rubric

Threat modeling deliverables (50% overall)

Part 1 – 25%

- Description of attack sequence
- Python script
- *.pcap file capturing attack
- Description of results from the attack

EN.650.624, Network Security, Spring 2020 Instructor: Reuben Johnston, reub@jhu.edu

Part 2 – 25%

- Details for installing Kerberos server, authenticated network users, and NFS/ssh servers authenticating using Kerberos
- *.pcap file capturing traffic between authenticated client and application server
- Description of advantages to this setup vs that in part 1

Part 3 – 25%

- Suricata installation and configuration details
- Reporting details from Suricata

Part 4 – 25%

- Discussion

Executive summary report (50% overall)

Table 1-Executive summary grading rubric (2006, Leydens, Santi)

Objective	**1 - Exemplary**	**2 - Proficient**	**3 - Apprentice**
Format / layout / organization	Report is very clear, coherent with excellent transitions	Report is clear and coherent, strong throughout	Report has some gaps, some weak sections
Writing mechanics	Report is virtually error-free, and contains few if any reader distractions	Report is logical and easy to read, and may contain a few errors causing minimal reader distraction	Report is generally clear, but distracting errors and flow make it difficult to follow at times
Figures / Tables	All figures and tables are easy to understand, and are clearly linked to the text. Story can be told almost entirely through figures.	All figures and tables can be understood with information given and are linked to text. One or more need improvement. May need more figures to tell the story.	Figures and/or tables are hard to understand, are not all linked to text. Several need improvement. Several more figures are needed to tell story.
References	All sources identified and referenced appropriately. Evidence of careful and thorough research for outside information.	All sources identified and referenced appropriately. Includes mostly readily available works.	All sources identified. Only readily-available works included. Some weaknesses in referencing, such as missing publisher information.
Typical Grade (average):	90-100%	80-90%	70-80%

References and useful resources

- https://github.com/jhu-information-security-institute/NwSec/wiki
- https://help.ubuntu.com/community/SettingUpNFSHowTo
- https://tools.kali.org/password-attacks/hydra
- https://recipeforroot.com/attacking-nfs-shares/
- https://ubuntu.com/server/docs/service-kerberos
- HW4 (for email server reference)
- 2006, Leydens, Santi, “Optimizing Faculty Use of Writing as a Learning Tool in Geoscience Education” (provided the rubric specified for the executive summary report)