

# Prepare



Connect with trusted communities  
to share information

# Prepare



Implement a central log management system

# Prepare



Implement event and alert management system for effective first triage

# Prepare



Implement case management system for effective incident handling

# Prepare



Implement automation and orchestration to support analysts during first triage and incident investigations

# Prepare



Implement detection use cases to cover the required scope

# Prepare



Build know-how for new threats and provide trainings to analysts

# Prepare



Establish visibility into all critical assets and platform

# Prepare



Ensure central log collection for relevant logs on systems in monitoring and incident response scope

# Prepare



Deploy remote forensics software

# Prepare



Deploy tools and capabilities for containment

# Prepare



Prepare decision tree and establish processes for GDPR readiness

# Prepare



Attend or organize internal and external exercises

# Prepare



Prepare and build coordination and escalation team for major incidents

# Prepare



Prepare and implement playbooks  
for assisting analysts

# Prepare



Prepare a list of critical assets for prioritisation

# Prepare



Ensure knowledge of all used operating systems in the team

# Prepare



Establish different kinds of communication channel for various stakeholders

# Prepare



Make a list of contact details for relevant peers

# Prepare



Ensure and implement collaboration between SOC and CSIRT and clarify responsibilities during an incident

# Prepare



Build backup infrastructure in case  
of emergency

# Prepare



Search for threats through regular threat hunting

# Prepare



Implement detection to find sensitive data which was indexed in a public search engine

# Prepare



Joker - invent a new preparation step

# Prepare



Joker - invent a new preparation step

# Prepare



Joker - invent a new preparation step

# Detect



Search for email attributes inside all  
of your mailboxes

# Detect



Search for cron jobs on Linux server

# Detect



Search for configuration settings on  
remote system

# Detect



Search for malicious configuration  
changes on remote system

# Detect



Search for processes on a remote host

# Detect



Search for files on a remote host

# Detect



Search for file hash on a remote host

# Detect



Search for destination IPs in  
network logs

# Detect



Search for source IPs in your  
network logs

# Detect



Search for domains in your network logs

# Detect



Search for referrer in your network logs

# Detect



Search for user agent in your  
network logs

# Detect



Search for URL in your network logs

# Detect



Search for leaked data in public  
search engines and paste sites

# Detect



Detect credential abuse by looking  
at login anomalies

# Detect



Detect credential leakage through monitoring external paste sites

# Detect



Detect leaked sensitive information  
on public source code repository

# Detect



Detect lateral movement with pipes

# Detect



Detect lateral movement with local admin

# Detect



**Detect lateral movement with  
credential abuse**

# Detect



Detect malicious process injection

# Detect



Detect leaked information through active cybercrime forum monitoring

# Detect



Detect unauthorized access on  
remote system

# Detect



Detect unknown domain  
administrator in your user  
management

# Detect



Detect unknown local administrator account on compromised system

# Detect



Detect in-memory malware on  
remote host

# Detect



Check GDPR relevance

# Detect



Find attacker through attacker's real IP which leaked because of bad op-sec

# Detect



Joker - invent a new detection  
method

# Detect



Joker - invent a new detection  
method

# Detect



Joker - invent a new detection  
method

# Respond



Remove persistence mechanisms  
on compromised system

# Respond



Remove malicious config settings  
on a compromised system

# Respond



Remove malware on compromised system

# Respond



Stop malicious process on  
compromised system

# Respond



Stop data leakage on compromised system

# Respond



Remove sensitive data on public  
accessible website

# Respond



Start first response to gather volatile  
data

# Respond



Collect and analyse logs from  
compromised system

# Respond



Collect evidence from compromised system

# Respond



Collect files from compromised  
system

# Respond



Collect the list of local accounts  
from compromised system

# Respond



Create and collect disk image for a compromised system

# Respond



Create file system timeline for a disk image

# Respond



Extract C2 IPs from a malware sample

# Respond



Search for found C2 servers

# Respond



Search for other affected systems  
after initial compromise

# Respond



Block domain names

Defensomania

# Respond



Block IPs

Defensomania

# Respond



Block URLs

Defensomania

# Respond



Block access to API

Defensomania

# Respond



Block employee account

# Respond



Isolate system

Defensomania

# Respond



Take remote system offline

# Respond



Disable network connectivity on  
remote system

# Respond



Delete malicious emails in  
employees mailboxes

# Respond



Sinkhole C2 server

Defensomania

# Respond



Capture traffic to C2 server

# Respond



Force password reset

Defensomania

# Respond



Add evidence to long time storage

# Respond



Mitigate a DDoS attack

# Respond



Initiate hot patching or immediate fix  
of exploited vulnerability

# Respond



Create a GDPR notification

# Respond



Document indicators of attack and indicators of compromise centrally.

# Respond



Share indicators of attack and of compromise with the community

# Respond



Joker - invent a new respond action

# Respond



Joker - invent a new respond action

# Respond



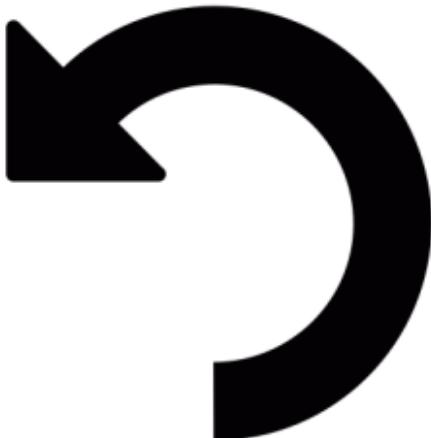
Joker - invent a new respond action

# Respond



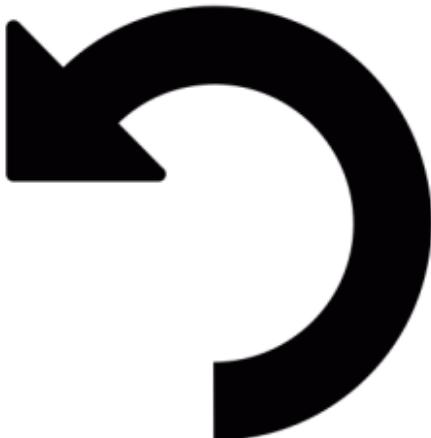
Joker - invent a new respond action

# Recover



Initiate fresh installation of  
compromised system from trusted  
install sources

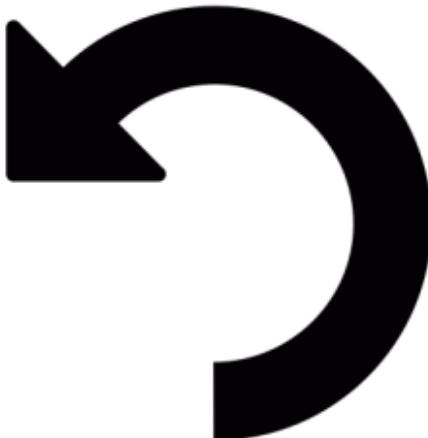
# Recover



Initiate the patching of a critical  
vulnerability

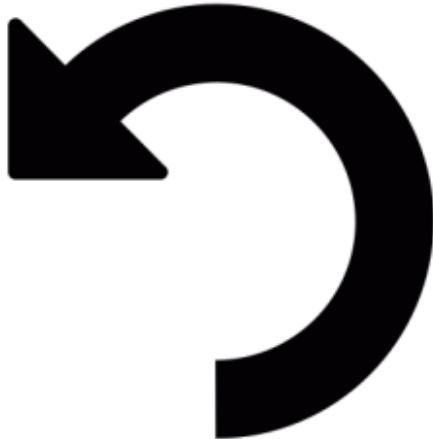
Defensomania

# Recover



Initiate an audit on a platform

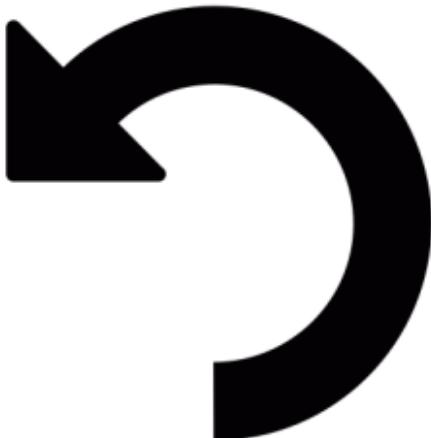
# Recover



Restore backup

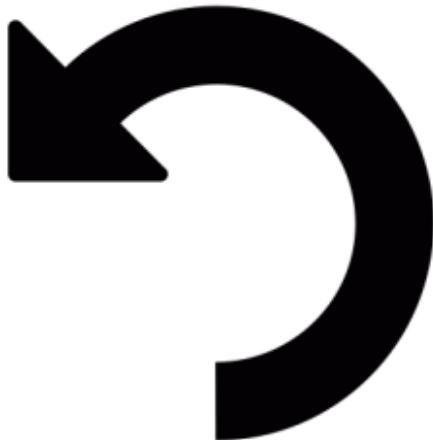
Defensomania

# Recover



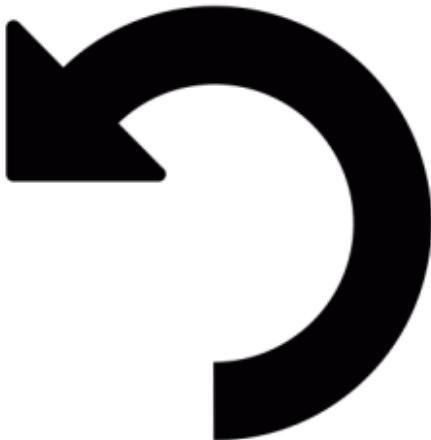
Bring the service online which you took offline during the incident

# Recover



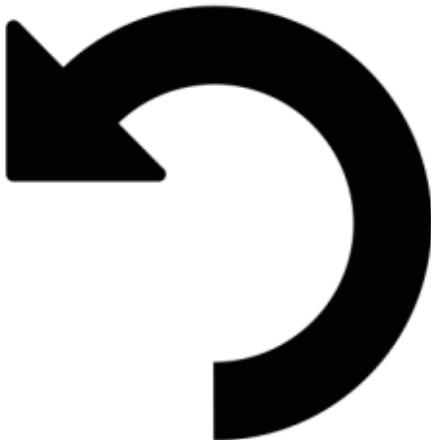
Rotate API keys for a compromised account

# Recover



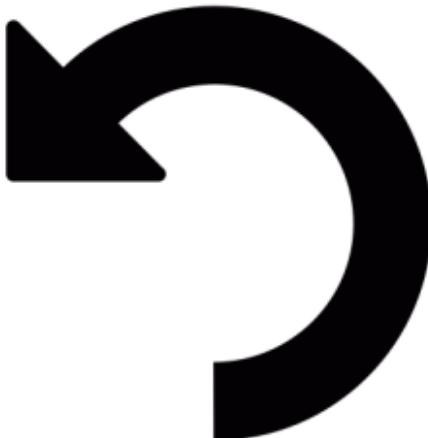
Change credentials for a compromised service account which was disabled during the incident

# Recover



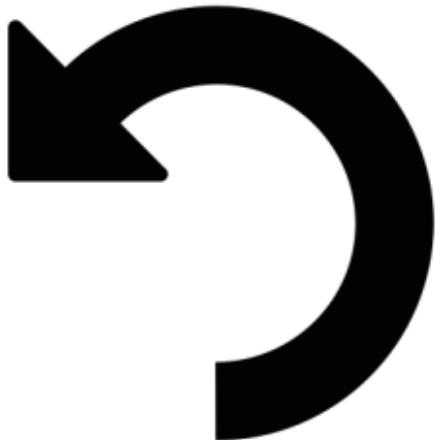
Implement long term monitoring  
through new security monitoring use  
cases and incident response  
processes

# Recover



Add a security deficit to the policy framework

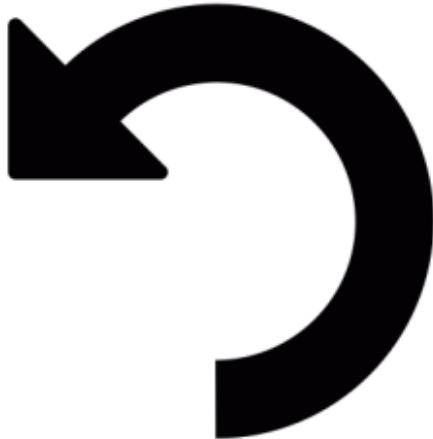
# Recover



Joker - invent a new recover action

Defensomania

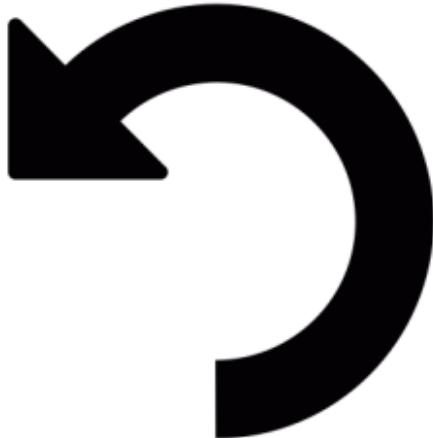
# Recover



Joker - invent a new recover action

Defensomania

# Recover



Joker - invent a new recover action

Defensomania

# Lessons Learned



Organize and perform lessons learned meeting

# Lessons Learned



Define technical debts and report them accordingly

# Lessons Learned



Document results and new knowledge for the team and company

# Lessons Learned



Improve incident response  
processes

# Lessons Learned



Improve incident detection  
capabilities

# Lessons Learned



Add new detection rule to internal  
and external repositories

# Lessons Learned



Joker - invent a new lessons learned step

# Lessons Learned



Joker - invent a new lessons learned step

# Lessons Learned



Joker - invent a new lessons learned step

# Lessons Learned



Joker - invent a new lessons learned step

# Communication



Publish external news article

# Communication



Inform public relations department  
about possible questions from  
outside

# Communication



Communicate findings to  
management

# Communication



Create an incident report

# Communication



Write answer to request made on  
external platform

# Communication



Joker - invent a new communication  
way

# Communication



Joker - invent a new communication way

# Communication



Joker - invent a new communication way

# Scenario



A supplier of your HRM software was targeted and a backdoored version was distributed. You may already upgraded. Indicators of compromise were published.

# Scenario



Your national CERT has published some indicators of compromise for your country and specifically for your industrial sector. Report back if you found hits.

# Scenario



Different employee accounts were abused for further attacks against your customers. During initial analysis, you found that all of them visited a typo domain of your corporate portal.

# Scenario



An employee's workstation was infected with a banking Trojan which not only changed network settings but also installed email monitoring software. It is related to a recent malvertising campaign.

# Scenario



You are notified that one of your ticketing systems is compromised and malware was placed on that server. It connects to other systems reachable from the ticketing server.

# Scenario



Spear phishing attack against an employee and employee reports login in into a similar looking company portal.

# Scenario



Spear phishing attack against an employee using a malicious document which after it was opened executed code on the computer.

# Scenario



Employee downloads a malware through a fake software package which connects back to command and control servers using DNS.

# Scenario



Vulnerability was exploited on one of your servers and an attacker had access to the database with customer data. A increased bandwidth was registered on the network device.

# Scenario



A database with customer data was exposed to the Internet through a misconfigured firewall and someone on Twitter is leaking data from that database.

# Scenario



A public git repository leaked internal credentials and it's unclear whether the credentials were already abused.

# Scenario



Servers from your main online service were targeted by a DDoS and are unavailable.

# Scenario



Malicious code was distributed to your endpoints during the routine update of a signed application.

# Scenario



A domain admin has run a malicious attachment after loudly proclaiming how dumb their users are for doing the same.

# Scenario



Security company has released an APT group PDF at BH USA. The list of IOCs includes a host in your ASN.

# Scenario



Every night new corporate accounts  
are abused for sending spam.

# Scenario



Email infrastructure got compromised. Attackers have full access to your mailboxes.

# Scenario



Attackers got domain admin in your environment. What should you scary more? Backdoor accounts or the sudden activation of disk encryption software?

# Scenario



An adversary has access to your Jenkins server.

# Scenario



An adversary has access to your vulnerable Jenkins server. Jenkins jobs with credentials are visible to everyone.

# Scenario



Someone is impersonating you to customer support at one of your service provider.

# Scenario



Your DNS was modified to respond  
with an attacker's DKIM key.

# Scenario



Due to a botched CI/CD script,  
complete source code exposure on  
production.

# Scenario



A bug in your webapp has allowed every record to be accessed via URL enumeration and IDOR.

# Scenario



Your subscription database is hacked. Thousands of new accounts are added and hard to distinguish from old.

# Scenario



New paste on a public paste site  
was found with password hashes  
and emails from your customer DB  
from 3 years ago. More recent  
hashes are missing.

# Scenario



Your reception software has leaked all of your visitor logs. They are available via torrent.

# Scenario



A script has been logging exported variables containing full user registration objects. This logging method is full of plaintext passwords.

# Scenario



Joker - be evil and invent a new nightmare and worst case scenario.

What scenario will get your team struggling? (If you want to get crazy then contribute your scary scenario for the sake of humanity.)

# Scenario



Joker - be evil and invent a new nightmare and worst case scenario.

What scenario will get your team struggling? (If you want to get crazy then contribute your scary scenario for the sake of humanity.)

# Scenario



Joker - be evil and invent a new nightmare and worst case scenario.

What scenario will get your team struggling? (If you want to get crazy then contribute your scary scenario for the sake of humanity.)

# Scenario



A developer has just typo'd an upstream package installation to their laptop. There was a malicious package waiting for that typo and post-installation code is exfiltrating data.

# Scenario



An engineer has typo'd a package  
they are adding to a product  
repository. It is malicious. The CI/  
CD and prod environment variables  
are exfiltrated to a C&C

# Scenario



A malicious browser extension was installed on different corporate computers. It injects keyloggers into websites.

# Scenario



Your build imports from an employee's personal NPM package.

They quit, and vandalize the package, causing a public incident.

# Scenario



An adversary takes control of your MDM service account. Your MDM support team cannot be reached for account recovery.

# Scenario



Your DNS was modified to respond  
with an attacker's DKIM key.  
Spearphishes will be signed by your  
domain in 30 minutes.

# Scenario



The certificates involved with your primary code signing process have been compromised and used to sign malicious apps.

# Scenario



An employee has left a firewall rule wide open after several hours of troubleshooting a network issue.

# Scenario



An employee has left a firewall rule wide open after a faulty change request was implemented.  
Bruteforcing attempts were registered on previously protected servers.

# Scenario



You allow customers to upload sensitive information to your platform to share them with others.

However, the data was left unprotected and anyone could access and download the content.

# Scenario



You were informed that one of your website directories used to share files with external parties were accessible and writable by any anonymous user. Suspicious files were found.