Feedbacks on 10y of pentesting and DFIR
How to increase your detection capabilities

Julien Bachmann
@milkmix_

OWASP, Geneva
24 April 2017
INTRO
ABOUT ME

Julien Bachmann

Current
CTO @ Hacknowledge
Swiss security monitoring solution

Guest lecturer @ Swiss schools on software exploitation and dfir

Past 10 years
• Security Researcher
• Security Architect
• Pentester and incident responder

OWASP, Geneva 24/04/2017
No magic spell
• Unfortunately what I present will not make you hackproof
• Still looking for the magic solution if anyone care to share ;)

Yet, techniques learned from the other side
• Before being fully on the incident detection side, we were mostly “creating them”
• Help to detect incidents faster
• Practical tips
Companies still got owned
• No-one found this silver bullet, yet

Mean time to discovery is still high
• Could be up to 6 months, or even more
• Problem is that attackers can grab their loot or destroy your infrastructure in less than a week...

Discovery is often not due to the company own detection capabilities
• Ransom request or public leak
• Third party that detected something suspicious
“Yeah but it’s those damn 0day! What could I do!”

- Unfortunately it’s not, stop blaming them
- Yet this could be a major PITA if the attack is targeted or event large scale 0day shopping
  - Struts2 CVE-2017-5638 at the beginning of this month
  - CVE-2017-7269

But true that they can hit you

- Worm using 0day to propagate
- Sometimes the patch is not existing yet
  - CVE-2017-0016
  - Still, most of the time it uses a N-day that hasn’t been patched yet

OWASP, Geneva 24/04/2017
FIRST STEP
DON'T FORGET
PEOPLE
Before speaking about technical aspects

- This part is often neglected and creates silos within the company
- Dev vs Ops vs Net, all against Sec ;)

- How to detect suspicious behavior in your business application if security never spoke with business people?
LEVERAGE PEOPLE

Methods that help

• Recruit security champions within teams as liaison-agents

• Join the DevOps/Agile movements and integrate security within all processes
  • Easier said than done

• Also, use techniques advertised by DevOps movement
  • CD/CI
  • API and integrate your tools
THE EXFILTRATION CASE

OWASP, Geneva 24/04/2017
Attacker compromised company’s infrastructure
• Gain access from vulnerable server in the DMZ
• Pivoted a few times
• Gain access to internal infrastructure

Their goal
• Data extraction
• Created massive tarball with files to extract
How attackers got detected?

- Windows administrator created alerts for hard disks nearly full, which triggered
- Inspected the machine and found the large file
- Listed processes and schedule tasks

>> Called the ghostbusters ;)

Morality

- Use monitoring tools as a first easy line
Next step: gain persistence

- Multiple ways to do so like registry keys, services, or...
- In this case they used Scheduled Tasks
- Tasks ran RAT dropped and stored locally

Services

- Another way to gain persistence is through services
From the blue team side of things this leaves plenty of traces!

**Execution**
- Execution of at.exe
- Creation of tasks pointing to suspicious folders

**Logs**
- Creation of a scheduled task : eventID 106
- Creation of a new service : eventID 7045
- RunAs generated by scheduled tasks 4648
- Traces of execution of the at command: eventID 4688
  - Since 7 / 2008r2, but you don’t have any XP/2003 left yeah?
  - In the GPO : Process Tracking > Process Creation
  - Don’t forget to enable command line traces

OWASP, Geneva 24/04/2017
Ok they got access and persistence now what?
• Multiply, just like Gremlins!
• Meaning: look for other targets to pivot to on your infrastructure

Techniques that can be used
• Basic network scan for 135/tcp and 445/tcp
• RDP or SSH scanner and bruteforce
• Larger range of ports to scan (ex: nmap in powershell)

“Advanced” attacker
• List connections using netstat command ;)}
Again, back on the blue team side

Network probing implies
• Connections to hosts that shouldn’t be contacted

Bruteforce implies
• Plenty of failed authentication attempts
• If you enabled those...

Good reason to use old friends that are quite hype lately
• Honeypots
• Blackholes that accept everything and throw alerts
Last step, they want access to files
- Will issue searched for interesting files
- Based on name, metadata and content

Probably not only on file shares but also on email accounts
- Trying to gather more privileges and access administrative interfaces
But the Blue team is still here watching!

Access to files can be detected
- Monitor specific files and folders using Windows Audit (eventID 4663)
- Create fake accounts and/or login interfaces
  - One reason why communication w/ business applications team is important

Deploy files that callback once opened
- Idea popularized by the OpenCanary project
Several ways to infect a machine

- What is considered “advanced”: exploit kits
- What is considered “low-tech”: social engineering

Everyone thought that macros problem was solved...

- Reality is we (security industry) spend too much time thinking about “advanced” vectors
- Way more fun than macros!
- Not really taking the problem to its core

Analysis tools focused on binaries

- Attackers switched to script languages
- Javascript and Powershell are all the rage lately
**THE MALWARE CASE**

Detection on the network side is limited
- IDS are like AV: based on signatures that can be bypassed
- But still really useful when properly tuned

Recent cases have made it even more so
- Let’s Encrypt and certificates for everyone
- Dridex campaign hosted on Azure Sharepoint
- Cerber campaign hosted on Dropbox
A common schema lately

- Credit: govcert.admin.ch

OWASP, Geneva 24/04/2017
When attackers want to bypass UAC

- Should be less of a problem in enterprises
  - You don’t have users w/ administrative rights right?!?
  - Social engineering still seems to be the most successful path

Leveraging logic flaws in Windows signed binaries

- Some executables from Microsoft allow to elevate privileges w/o UAC prompting
  - Usually patched by Microsoft
From a Blue team perspective several scenarios

“We block all macros and Powershell scripts using it’s execution policy”
• Or for the more startup-ish: “we use only cloud-based editing suites ;)”
• Also possible to block activation of macros downloaded from the Internet through GPO

Problem is that it is rarely deployed
• Operation charge is too costly in most cases
• Business workflows requiring macros for example
• As for Powershell, it is possible to bypass execution policies
  • Up to v5
Taking a logs and detection approach

“Standard ops w/ wscript.exe and powershell.exe processes ran from Word.exe?”
- Need to study the attacker (cyber-kill-)chain
- Start with easy rules based on parent process
- Add processes command line

“ok, I will buy that EDR. <SecConfXYZ> had a floor full of them”
- Not so fast, actually Microsoft got you covered in this area
- Out-of-the-box since 2008r2 and getting better since last year!
Audit processes creation from the GPO
- EventID 4688
- Don’t forget to enable command line from Server 2012r2

SysInternals Sysmon
- In short: Microsoft free EDR
- Well almost... only the reporting no analysis or correlation is made out-of-the-box
  - Except if using Defender Advanced Threat Protection
  - Sadly, it is cloud-only...
- Simple to configure
- Public large deployments documented to reassure you
The rest is up to you: create detection rules by knowing attacker’s techniques

- Suspicious parents for set of applications
- Suspicious children for set of applications
- Suspicious execution paths for applications
  - %APPDATA% for example
- ...

Powershell examples

- Detect “-Version 2” in command line
- Argument that looks like base64 encoding
- Detect “-EncodedCommand“ argument
- ...

Audit what is executed on your infrastructure
• And disable macros and executables ran from users writable folders
Or if less lucky...

- The APT case
- Only studying a few techniques for this talk due to time constraints

Once they obtained administrative privileges (left as an exercise)

- Trying to retrieve passwords from the memory
- Generate magical Kerberos tickets

Persist, access data and be stealth

- Adding themselves to specific groups (Domain Administrators, RnD, ...)
- Erase logs to prevent analysis
Even with more advances attackers, its possible to do something
- Mimikatz has some specific execution patterns
- Access to lsass.exe process to 'ask nicely' for credentials
- ...

Sysmon logs
- Access to lsass.exe w/ AccessGranted set to 0x1410 or 0x1010 since last release
- ...

Microsoft Advanced Threat Analytics
- Attacks such as Pass-the-Ticket and Skeleton keys have specific behavior
- Need logs and network view
- This tools from Microsoft automates the analysis 😊
**THE PENTESTER CASE**

*Watch out for events on specific objects from the Active Directory*
- Modifications in groups: eventID 4728
- Create filters based on your environment

*Logs cleared*
- Fortunately you have everything centralized in a SIEM
- EventID 1102

https://speakerdeck.com/milkmix/import-module-incidentresponse
Plenty of other techniques can be used

- Access to Linux hosts
- Exploitation of vulnerable services
  - Struts2
- Exploitation of misconfigured environments
  - JBoss anyone?
- ...

OWASP, Geneva 24/04/2017
Same for the Blue team, plenty of interesting topics
- Office documents analysis using oletools, mraptor, ...
- Endpoints analysis using tools such as sysdig, GRR or osquery
- ...

Don’t forget your cloud environments
- Use logs provided by the platform
  - Ex: CloudTrail and CloudWatch on AWS

Unfortunately, no time to cover everything in this talk
- Available to discuss techniques and cases
- Don’t hesitate to reach out

IN CONCLUSION
Approach to follow?
In fact, no need to log *everything* but better not miss things required afterward

**Be smart and study attackers techniques**
- Have a lab to try them and study side-effects
- Review your pentest reports with your sysadmins
- Read detailed attacks analysis in specialized blogs and transform actions into logs

**Review Microsoft documentation and SysInternals tools**
- Really improved the last 10 years
- No need to buy shiny tools all the time
- Better use time to learn to use the provided tooling

*Choose security tools that can easily be integrated in larger workflow*
Approach to follow!

API
ALL THE THINGS
Look at all recent security tools in the open source side of the fence
• All provide API to better integrate with others!
• Standalone products are limited or you need the full package from $EDITOR

As said in intro, integrate your tools
THANK YOU!

QUESTIONS?

See you soon