Learning from the attackers

What's the attacker teach us on how to improve our information systems ?

Alexandre Dulaunoy

ASBL CSRRT-LU (Computer Security Research and Response Team Luxembourg)
http://www.csrrt.org/

May 9, 2008

Introduction or Disclaimer

- After a deep analysis of the data captured in a honeypot, we discovered a lot of attackers tools and tactics (sometimes their motives)
- What could we learn from such information ? Can we build conclusion recommendations about security ? or is it so empirical or focused that we'll recommend wrong paths at the end ?
- At least, we'll make a partial bridge between theory and practice in "computer/network" security

Terminology : users are running information systems and attackers are the one trying to attack them.
An user can become an attacker and an attacker can become an user

Attackers exist

- There was an old myth : "There is no such thing as attackers in cyberspace"
- They exists and looking at the kind of interaction within the honeypot, they are human
- Thinking and implementing security on your information system is not worthless
- A collateral law, they often don't target countries but just the cyberspace and its big potential of vulnerable systems

Attackers discover and exploit vulnerabilities

- Attackers use known or unknown vulnerabilities
- Attackers often use vulnerabilities before their are falling into the known category (e.g. ssh exploit, ptrace bug,...)
- Protecting information systems on only known vulnerabilities is just covering a part (e.g. IDS or malware patterns)
- Implementing Least privilege is important to better contain unknown and known vulnerabilities ("'The principle of least privilege states that a subject should be given only those privileges that it needs in order to complete its task"')

Attackers discover and exploit vulnerabilities (2)

- In recent web security issues, the permissions on the /tmp or /temp directory are very important and must follow least privilege principle
- Implementing Fail-Safe principle in software and in their implementation (in other words "' In doubt, a none access is given."' or avoid default allow from software to network configuration.)
- Attackers monitor security advisory and you ? (e.g. RSS security feeds are nice and free, they just need a bit of time)

Attackers innovate

- Some years ago, we already discovered a lot of attackers using tunneling protocols like ipv6 over ipv4 to just hide their activities
- Attackers innovate just to pursue their objective
- Don't minimize their ability of adaptation
- Innovating don't mean "buying bleeding edge devices" but more "what are the (bad & good) potential use of a technology ?"

Attackers innovate (2)

- The innovation process of an attacker is often a "thinking out of the box approach" :
  - User trying to protect their system : "What are the risks ?"
  - Attackers trying to attack a system : "How to attack the system ? nice the service X or Y is running, I'll give a try"
  - Each perspective are valid but sometime user should take the hat of an attacker against their own system
  - It will better refine the risks and where to focus

Attackers communicate

- Attackers communicate just like human
- Attackers communicate (very) well :
  - They will use any channel of communication available including covered channel (e.g. from IRC to ICMP tunnelling)
  - They exchange information with other attackers and non-attackers (e.g. the announce of a compromised)
  - They often integrate the system in a larger network of compromised systems (e.g. IRC interface to your own credit card verification process)
- and you ? are you communicating with colleagues, suppliers or competitors having the same security troubles ?

Attackers communicate (2)

- When implementing access or remote services, think twice before enabling it as it will be used as a communication channel by the attackers :
  - Does the machine in a DMZ really need an Internet access ? often it's not required and helps the potential attackers to communicate (e.g. using the system to launch other attacks, download toolbox like root-kit, being part of a larger network of compromised system, ...)
  - Don't forget a lot of protocols are full-duplex and encapsulation of non-legitimate traffic is possible (and often easy)
  - A end-user can be a communication layer without knowing to be one (e.g. p2p protocol)

Attackers have toolboxes

- As seen in the Honeypot, attackers use toolboxes to ease their work
- Sometimes they compile or execute their toolboxes on your compromised machine :
  - Do you really need a C compiler on your machine ? do you need a C# virtual machine or web browser on a server ?
  - Follow the rule of "if you don't need it, don't install it or remove it"
  - If an attacker is able to install software, are my permissions correct ? do I follow the principle of least privilege ? separation of privilege ? do you control regularly file integrity ?
- and you ? do you have your toolbox to analyze a compromised system ? and do you know how to use it ?

Attackers love any services

- A common myth in information security : "I'm using a so obscure protocol that no one is interested"
- Some experiment in Honeypot with obscure protocol shows an interest and exploit from the attackers (e.g. the mbus case)
- User often forget that the cost of testing large set of information system on Internet is low
- User must apply the "Principle of Open Design" ("'The principle of open design states that the security of a mechanism should not depend on the secrecy of its design or implementation"')

Attackers share

- Attackers share :
  - Compromised systems with other known (sometimes with unknown) attackers
  - Toolboxes and idea with the other attackers (e.g. root-kits are often trojaned by others attackers)
  - Services available on your nice compromised systems (e.g. a CC validation system)
  - and extend your compromised with new services (e.g. from a sniffer to a phishing website)
- Monitoring your systems for detecting not-known services is important
- Do you share tips about security ?

Attackers are sexy

- We don't know... no experiment where done until now
- Collateral point : When analyzing something on a compromised system, everything is a perception
- We learn everyday and security is a never ending process

Bibliography

- Know Your Enemy, The Honeynet project - various, (second edition) Addison Wesley, ISBN 0-321-16646-9
- Computer Security, Art and Science, Matt Bishop, Addison Wesley, ISBN 0-201-44099-7

Q and A

- Thanks for listening.
- adulau@foo.be