ICE
TRUSTED IT PARTNER

LITTLE BLACK BOOK OF CYBER ATTACKS

LITTLE BLACK BOOK OF CYBERATTACKS
TABLE OF CONTENTS

SOCIAL ENGINEERING

SOCIAL ENGINEERING

SOCIAL ENGINEERING

Social engineering is the art of "hacking the human." If a strong password secures a system, it is a lot faster to trick someone into giving you the password than using a password cracker. It is easy to make fun of the idea of being tricked into revealing your password. However, this sort of mocking is why social engineering is so effective. Social engineering takes advantage of the cognitive biases and social norms that we grew up with and uses those biases and norms to manipulate us.

The Principles of Social Engineering

Like hacking, social engineering has core principles applied in various ways to manipulate a target. While the core principles in hacking exploit the design of our technologies, the principles used in social engineering exploit our cognitive biases and social norms.

Principle 1: Consensus

Consensus, also known as Social Proof, is when a social engineer convinces victims that others have already trusted them. There are many ways to apply consensus to social engineering. Some examples are fake website reviews on social media, forged work orders, using deep fake technology to impersonate someone over the phone in real-time.

Principle 2: Familiarity

Familiarity, also known as Trust, refers to a social engineer using charisma and likeability to get a victim to complete a request. Familiarity is often as simple as striking up a friendly conversation with a victim before making a request. It is often even more effective if the social engineer is, or appears to be, part of the victim's "in-group." For example, striking up a conversation with a security guard about the struggles of the night shift or manipulating an inexperienced receptionist by pretending it is your first day and you forgot your badge.

Principle 3: Urgency

Urgency, also known as Scarcity, refers to a social engineer creating a sense of hurry to put time pressure on a victim. Creating a sense of urgency discourages the victim from thinking critically about the request while also making them feel like they are helping someone in need. Urgency is a time-honored tradition among scammers. One example is emails and robocalls informing you that your warranty is about to expire unless you ACT NOW!

Principle 4: Authority

Authority, or Intimidation, is a high-risk strategy in which a social engineer attempts to intimidate a victim or claim authority over them. While usually more subtle than "Don't you know who I am," Intimidation is not a subtle strategy. It has an elevated risk of the victim reporting the incident rather than complying. More practically, Authority can be combined with Consensus to create the illusion that not only is the social engineer already trusted but that they are trusted by someone with authority over the victim.

Conclusion

Social Engineering is a very effective tool used by ethical and malicious hackers alike. While strong technological security is important, ignoring the human aspect of security creates a severe vulnerability that threat actors can easily exploit. Protecting against social engineering is more than annual training; it requires creating a security culture where people understand their importance and the threats they may face.

PHISHING

Different Types of Phishing

Social engineering is a common tactic used in all forms of phishing to persuade victims to act, however, there are other techniques and targets outside email:

Vishing: which is short for "voice phishing," describes spam calls in which a caller poses as a representative of the victim's bank or police enforcement in an effort to obtain personal information.

Smishing: is when an attacker sends a malicious link through text message, they are said to be "smishing," short for "SMS phishing."

Whom it targets is another way to classify phishing. In an effort to catch a victim in a large net, many phishing efforts send out bulk spam emails to people and organizations. However, there are situations when an attacker designates a specific target and emails that target specifically. Spear phishing is what this is. Whaling is used when the target is highly sought-after, such as the CEO of a firm.

Phishing uses humans as its initial attack vector, whether it's employed to mislead a victim into paying money, collect login information, or download malware.

How Does Phishing Work?

Phishing attempts can occasionally be as simple as emails or phone calls that ask the victim to give money or personal information to the attacker. Others call for much more technical skill, including those that persuade users to click on links that download malware onto their devices. For instance, an attacker may socially engineer a user into downloading and opening a PDF or Word document that has malicious code included in it or attach it to a phishing email.

The ability to spread the infection by sending more phishing emails to the user's contacts is frequently included in this harmful code.

Email Spoofing

When an attacker forges email headers to make it appear as though the email is coming from someone else, this is known as email spoofing. Up to 90% of email fraud assaults employ spoofing, which is a prevalent component of phishing emails.

Normally, the "from" field is already filled up when you send an email. My acquaintance will be able to tell that I sent an email to them if I use the email address john johnson[@]gmail[.]com. But you can also send emails using straightforward scripts (here are instructions for sending an email in Python).

When you write and send an email using a programming script, you can configure the email headers to be whatever you want - meaning that an attacker can put any email as the “sender”, even yours.

In order to really see what is going on in an email, you can download it and open it in a code editor, but most email providers allow you to see the email headers from within your email. For instance, with Gmail, you may view the email headers if you open an email that piques your curiosity, click on the three vertical dots in the top right corner, and select "Show original."

These email headers provide crucial data that can be used to identify phishing, such as the "return-to" address, sender IP, and whether any anti-spoofing measures like SPF and DKIM were ineffective (they are the reason emails are automatically sent to your spam folder). Before responding to a suspicious email, it is always advisable to read the headers to check for "failed" protection fields and to see the original sender's IP.

Not Just Emails: Webpages That Steal Your Password

Particularly successful phishing tactics are websites that collect login information. The user is unaware they were phished because these pages frequently direct users to legitimate websites after collecting their login information. These websites may also tempt you to unknowingly download malware.

A typo-squatting domain like iceconsulting.cm or icconsulting.com could be used to steal ICE Consulting logins if a user types in the erroneous domain on purpose. To entice someone to click through to a hidden domain, a malicious actor could potentially mask their domain with a link shortener like bitly.

Conclusion

Phishing is a challenging issue to deal with because to the wide range of phishing kinds, the low cost to construct phishing pages, and the simplicity with which one can do so. Furthermore, no system in the world can guarantee against a human employee clicking on a bad link, regardless of how sophisticated it is. As a result, it's critical to report any questionable emails or links to the relevant department at work so that they can block the senders and sites. One person can do a lot to defend a business and oneself against phishing attempts if they pay attention to the little things and report questionable content.

SOCIAL ENGINEERING OF EMAILS

Spam, or sending unsolicited emails, is a very successful social engineering tactic. Most spam emails that land in our inboxes is blatantly false, and this is done on purpose because the scammers who send them are looking for easy victims who lack the intelligence to spot schemes. Even though fewer individuals will open the email, those who do are more likely to fall for a scam.

In contrast to these con artists, social engineers frequently utilize spam that is designed to be difficult to detect to bypass spam filters and appear authentic. Most people are aware that they shouldn't believe emails that look to be from their company's IT department, but what about emails from odd dating sites we didn't sign up for? These emails frequently prey on our trust by pretending to be from reliable sources, and the practice of "prepending" can make matters worse.

Prepending is the process of changing the subject line of an email or adding a message that reads "RE:" or "MAILSAFE:PASSED" to the body of the message to make it seem as though: We have already been in contact with the sender; OR The email has made it past a spam filter.

When done well, this might give the unwary victim a feeling of even greater security.

Hoaxes

A fundamental component of social engineering, lying to get what you want can take many different shapes. It frequently fits with one or more of the fundamental tenets of social engineering. These lies are frequently referred to as hoaxes in social engineering. A typical hoax is to pretend to be security alerts to make the victim feel rushed. Because the alerts seem to be from legitimate sources and direct the victim to follow instructions, these phony alerts frequently abuse the victim's sense of authority and trust.

Pretexting is another type of lying employed in social engineering, in which a social engineer creates a fake justification for why a victim should divulge information or take an action.

You would probably disregard someone if they sent you an email asking for private information. On the other side, you can be duped into disclosing information to them if they claimed to be the new point of contact for a contractor working for your company.

Social Engineering and URLs

With the internet, deception is now simpler than ever, and it can happen even before a consumer visit to a website! A straightforward link to a "trusted" website can be used to deceive victims by social engineers.

When a social engineer guides people away from a trustworthy website and toward their harmful website, this tactic is referred to as pharming. Typically, this entails altering DNS data for a machine, a network, or a broader area of the internet. Making the name resolution procedure point to a different IP address enables phishing. This is frequently used to obtain banking credentials from gullible victims.

Typosquatting is another technique used to seduce unwary people into visiting harmful websites. Typosquatting is the practice of an attacker registering a domain that is strikingly like an already-existing, trustworthy website, then watching for users to access the malicious domain. For instance, a hacker might register codeAcademy.com to deceive users who are trying to access the website. A mistake as easy as mistyping or forgetting a URL could lead victims to this malicious domain.

If you're skeptical of this strategy, try finding the differences between these URLs:

kerning.com vs keming.com

google.com vs goggle.com

Identity Fraud

When an attacker utilizes a victim's personal information, it is called identity fraud. Many of us have heard of instances where dishonest individuals have pretended to be someone else to profit. Using someone else's bank account or credit card, for instance.

The purpose of identity fraud is not always financial gain. Social engineers can also use it to pose as a victim more effectively, either to deceive others or to obtain more access to the victim's accounts and resources.

On a personal level, this can involve targeting a company that the victim works for or utilizing stolen personal information to "recover" a bank account.

Larger-scale instances of this type of fraud might take the shape of invoice scams, in which an attacker modifies an invoice's details to steal money. Using social engineering to pretend to be an employee of one company and submitting false invoices to other businesses is one type of invoice scam.

If the attacker is successful, the second company won't notice the issue and will just pay the invoice.

Credential Harvesting

When an attacker obtains, or harvests, a victim's credentials, this is known as credential mining. However, credentials are frequently taken from numerous users at once, usually for financial benefit. This can be aimed at a specific person as part of a multi-stage attack.

A watering hole attack is one technique for obtaining specific credentials. An attack known as a "watering hole" occurs when an attacker gains access to a target by compromising a service, piece of software, or website used by a third party. The "watering hole" from which all the victims are "drinking" is the third-party service. This is an illustration of how shoddy security practices by third-party providers can jeopardize the safety of the companies that use them.

In 2012, a hacker gang attacked websites that supported political activism as an example of a watering hole assault. Attackers attempted to install malware on the targets' PCs by leading victims to a different infected website.

Physical Social Engineering Strategies

For an attacker, having physical access to a target opens a world of new opportunities, and often getting physical access is simpler than using technology to get in. These methods focus on getting over physical security or gaining credentials in person rather than online.

Shoulder surfing, the act of looking over someone's shoulder while they type their password is referred to as shoulder surfing. If the social engineer can do it without being discovered, this method of collecting credentials is quite effective, albeit it does require some skill on their behalf.

Dumpster diving is, as the name implies, the practice of searching through rubbish to find private information. Although it may seem absurd, this happens more frequently than you may imagine. Sensitive documents are frequently incorrectly disposed of by organizations, leaving them accessible to social engineers. This method can be used to access a variety of data, including sticky note passwords, employee data, and tax invoices. Don't forget to destroy your vital documents!

Tailgating describes the practice of following someone through a locked door before it closes. (“What? That qualifies as a cyberattack, you ask? Yes, it really can be that easy.) invoice.

Conclusion

Email social engineering is becoming frequently common and although some are obvious others are not. Many can spoof your employer’s email, legitimate institutions, or friends, family or coworkers. Telltale signs are being asked to change your password, provide any personal information, or requests for gift cards. When in doubt do not engage or responds with the email but contact the “alleged” sender for verification.

TERMS

- **Spam:** unsolicited emails.
- **Prepending:** attaching a message to an email saying something like "RE:" or "MAILSAFE:PASSED" to make it appear that the email is safe and legitimate.
- **Hoaxes:** fake information, like false security alerts.
- **Pretexting:** when an attacker tricks a victim by giving a false pretext, or reason, for why the victim should share information with the attacker.
- **Pharming:** when an attacker redirects victims from a legitimate website to their malicious version.
- **Typosquatting:** when an attacker deliberately registers a website domain with a name that is close to that of a legitimate website.
- **Identity Fraud:** is when an attacker uses a victim's personal information.
- **Credential Harvesting:** when an attacker is attempting to harvest or learn, a victim's credentials.
- **Watering Hole Attack:** when an attacker hacks the third-party service or software a group of victims uses to gain access to a victim or the victims' company.
- **Tailgating:** when an attacker follows someone through a secure door before the door can close.
- **Dumpster Diving:** when an attacker goes through a victim's trash to obtain sensitive information.
- **Shoulder Surfing:** when an attacker looks over someone's shoulder as they type their password.

INFLUENCE CAMPAIGNS

Large-scale initiatives called influence campaigns aim to sway public opinion. Such tactics frequently aim to spread a misleading narrative and are typically conducted with evil intent. Groups with high degrees of competence, up to and including nation-state actors, frequently carry out these activities.

How Do Influence Campaigns Work?

Social Media Manipulation

It is now considerably simpler to persuade vast numbers of individuals thanks to social media. They offer targeted advertising services in addition to being an easily accessible online social network. These services enable organizations undertaking influence efforts to target the people more precisely they want to persuade. By making it seem as though other members of the public are voicing the opinion, strategies like astroturfing, in which an influence campaign is camouflaged as a grass-roots movement, are used to sway public opinion.

In Combination With Other Tactics

Influence campaigns can be a component of larger campaigns that employ additional tactics like espionage and hacking. Such campaigns typically have a broad objective and employ a variety of strategies to strive towards it. In these situations, the influence campaign may serve more as a diversion or cover for other, more aggressive tactics rather than directly achieving the aim. Additionally, they might portray a target group as an enemy, which might lead to more direct acts being justified or ethically acceptable in the eyes of the public.

Conclusion

Influence campaigns signify a perilous new cybersecurity paradigm. They have a strong impact on public perception and are secretive and negotiable. Influence campaigns are additionally utilized in so-called "hybrid warfare," which combines conventional military tactics with hacking and the influence activities.

MALWARE, PASSWORD, PHYSICAL ATTACKS

MALWARE,
PASSWORD,
PHYSICAL ATTACKS

MALWARE

Virus

You go to your client's email and open it. You may tell right away that your client has opened certain emails coming from strange email addresses. When you open the emails, you can see that the client probably downloaded files and clicked on links in the dubious emails. Uh oh. Has your client downloaded any malware?

A harmful self-replacing program that affixes itself to other programs and executables without the user's consent is known as a virus. Data on the computer might be altered or deleted by a downloaded virus.

If the virus was able to access or alter data, the confidentiality and integrity of that data is now in question.

- Just like with adware, avoid suspicious links and install trustworthy antivirus software.
- Immediately report suspicious emails to your IT department and never open them.

Trojan Horses

There is another kind of infection that is not spyware, even if the existence of spyware makes it clear something malicious was placed on the computer. it's a Trojan Horse.

While the Trojan Horse, commonly just referred to as a "Trojan," is like Spyware in that it monitors activity on a system, it also performs other functions. Trojans are a sort of contained, non-replicating malware that impersonates trustworthy programs to gain access to a user's system and commit fraud. This malware infiltrated your client's PC by posing as a trustworthy antivirus program, just like the Greeks did when they crept into the city of Troy inside a huge wooden horse! Just like the Greeks did when they snuck into the city of Troy inside a massive wooden horse, this spyware entered your client's PC by pretending to be a reliable antivirus tool!

Spyware

Spyware is malicious software that users unknowingly download and use to steal personal data and transmit it to other parties in ways that are harmful to the original user. A threat actor might be able to obtain private data if the spyware included a keylogger, a tool that can record what a victim writes onto their computer.

This implies that any private information, including passwords, will soon be in the hands of an evil third party. Although spyware is typically not used to alter data, it nonetheless violates the confidentiality principle. It's possible that a hostile actor watched your customer type confidential information.

- Be careful what you click on and install that trustworthy antivirus already!

Worms

This virus is what kind? When you look through your client's "Sent Emails" folder, you discover that they recently sent the same email to all of their contacts. The emails are malicious, and the subject lines are identical. The email nearly looks to have copied itself...

You might have discovered a worm as opposed to a virus, which needs a file or application to spread.

Self-replicating software, or a worm, copies itself automatically from computer to computer. This worm might be as harmful as a virus.

The worm might also reproduce innumerably to the point where it overwhelms your client's system. The worm may disrupt the system and breach availability if it did this.

- Follow the previous suggestions for adware and viruses.
- Monitor the computer for any unexpected changes! Is it slower than usual? Is there less hard drive space than expected? Have files mysteriously appeared or disappeared? These could all be signs of worms.

Adware

You start by launching the web browser. It opens to an odd page promoting a computer cleaner that is "guaranteed to make your machine perform 10X quicker!!" on the first visit. What a strange selection for a homepage.

You observe several advertisements appearing everywhere as you browse the internet. Your screen is being overrun with them to the point where the website is actually loading more slowly.

This machine obviously has adware. Adware is unwanted software that is made to bombard your screen with advertising. Although not particularly nasty at first glance, adware occasionally is attached to other, more dangerous software.

This can actually affect performance if there is enough adware on your computer.

- Make sure to not click on any strange links or download any untrustworthy files.
- A trustworthy antivirus software could also help with this issue.

Fileless Malware

Fileless malware is a sort of malware that "lives off the land" and employs reputable programs and the operating system of the user to carry out harmful operations like privilege escalation, information gathering, and other things. Antivirus software nearly never picks it up since it is so difficult to detect.

In contrast to a Trojan Horse, fileless malware is a component of legitimate software rather than acting as if it were separate from it. Fileless malware blends in with normal software's code, frequently changing the existing code to make it dangerous.

These assaults are particularly susceptible to certain software, such as Microsoft PowerShell. This attack vector could be used by someone to obtain information, install malware, or mine bitcoin using the resources of your device.

- Did you download that antivirus yet? Still, avoiding those suspicious links?
- Disable command-line applications and macros, not in use on the device.
- Keep your applications and system up to date for the latest security updates.
- Reboot the computer.

Rootkits

What is the Trojan Horse doing exactly? What did it strive to accomplish? You must ascertain the solution.

When you scan the device, you discover that it is an awful device that keeps becoming worse since a rootkit was installed on the system via a Trojan horse.

A group of harmful applications known as rootkits allow unauthorized users to silently maintain privileged access to a system. Using a rootkit, a hacker can gain access to a computer by opening a backdoor. This rootkit managed to get administrative access to the machine, and it would be quite challenging to get rid of.

In this instance, the Trojan Horse impersonated a reliable antivirus program in order to set up a rootkit. This indicates that this computer's files are accessible by an evil third party who is located someplace. The confidentiality and integrity of your client's system are in grave danger in this circumstance. A rootkit can be removed with some specialist tools, but it is difficult.

- Back up any important data on this system and reimage it.

Ransomware

Someone had access to this machine thanks to the rootkit. With such access, what did they do? You notice that the rootkit was utilized to prevent the user from accessing system files that house a significant amount of crucial firm info.

Ransomware may be present if the bad actors restrict access to data or make threats to make the private information public unless the client pays them money. As threat actors have recognized it's safer and simpler to rob a virtual place rather than a physical one, the use of ransomware has been soaring! One of the biggest cybersecurity dangers now facing businesses is ransomware.

Data availability is seriously threatened by denying a user access. Even though availability might not seem crucial, for some firms, it can be disastrous. Imagine losing access to a hospital's system or a flying system's data!

- Regularly back up important files.
- Have a procedure in place for ransom requests. They should include a step in which the authorities are alerted.

TERMS

- **Malware:** Malicious code inserted into a system to cause damage or gain unauthorized access to a network
- **Adware:** Unwanted software designed to throw advertisements on your screen
- **Virus:** A malicious self-replacing application that attaches itself to other programs and executables without the permission of the user
- **Worm:** Self-replicating code that copies itself from computer to computer without user intervention
- **Spyware:** Malicious code downloaded without a user's authorization which is then used to steal sensitive information and relay it to an outside party in a way that harms the original user
- **Trojan Horse:** A type of contained, non-replicating malware that disguises itself as legitimate software to allow scammers and hackers access to a user's system
- **Rootkit:** A collection of malicious programs that secretly provide continued, privileged access to a system for an unauthorized user
- **Ransomware:** Malicious code that will block a user's access to data or threaten to publish sensitive data until they pay money to the malicious actor
- **Fileless Malware:** A type of malware that 'lives off the land' and uses legitimate tools and the user's operating system to perform malicious activities like privilege escalation, data collection, and more. It's incredibly hard to detect and almost always missed by antivirus software

PASSWORD ATTACKS

Password Terminology

Authentication and Authorization

Authentication and authorization may sound alike, but there is a crucial difference between the two: While authorization concerns a person's rights, authentication focuses on establishing your identity. As an illustration, logging into a computer is authentication, but whether you are permitted to run a certain program after logging in is authorization.

Authentication methods frequently involve passwords.

Hashing

Although hashing and encryption are linked, the main distinction between the two is that hashing cannot be reversed while encryption can. Hashing can be compared to a cryptographic mixer: It is mathematically impossible to reverse the technique and get the original data after receiving a hash, albeit there are methods that don't need doing so. Data is inputted, and a hash is produced based on that data.

Plaintext

Data that is stored in an unsecure, readily readable format that is NOT encrypted or hashed is referred to as plaintext. This can refer to plain text, but it can also refer to any unencrypted data, including images, movies, etc. For instance, "password" and "qbttxpse" are two distinct ways of representing the same data, but the first one is in plaintext and the second one is (very insecurely) encrypted.

Plaintext password storage is a bad idea.

How do you store the passwords?

Plaintext

Keeping the passwords in a plaintext file and ensuring that only you and the authentication system have access to it is the simplest and most obvious option. Theoretically, this would be a good way to safeguard passwords...

However, all the hacker needs to do to succeed is gain access to the file. There are numerous approaches to take, ranging from escalating privileges to just fooling the system into accessing the file on their behalf. Regardless of how they go about it, they already have the file and the unencrypted passwords. Clearly, a better answer is required...

Encryption

You choose to encrypt the password file after making a lesson out of the failure of plaintext storage. In this manner, even if the hacker is successful in obtaining the file, they won't be able to view the passwords! Right, the issue is resolved.

Not quite, I guess. If the hacker can gain the file holding the passwords, they will probably also be able to obtain the file containing the encryption key. Therefore, the encryption key must be stored somewhere such that the authentication system can access it.

Hashing

Since encryption didn't work, perhaps hashing—another sort of cryptography—can be of assistance. You wouldn't benefit from hashing the entire password file; but you can hash each password in turn. The user's password is then hashed and compared to a previously stored hash when they enter one. The password was accurate if the hashes matched. There must be no method for the hacker to obtain the passwords at this point.

Well, kind of... If you employed a secure hashing technique, the hacker could only determine the original passwords by repeatedly passing different character combinations through the same hashing algorithm until they obtain a result that matches the hashed password. A brute force attack, in which the attacker attempts a, then b, then c, etc., is the simplest method for accomplishing this. They move on to a, ab, ac, etc. once they've tried every character.

Unfortunately, this is rather slow, especially for long passwords, which is bad for the hacker (where it can take thousands of years or more). Unfortunately for you, the hacker previously obtained two lists of passwords in plaintext by stealing them. They can attempt going through the list and hashing each individual password rather than just guessing at random. Although it's unlikely to succeed in breaking all hashed passwords, there's a strong probability that it will. A dictionary attack is like a more sophisticated brute force attack.

Additionally, they don't have to decipher each hashed password individually: The hash connected with each account will be the same if more than one user uses the same password, allowing the attacker to create a large list of strings and associated hashes known as a rainbow table. Now they can save time and effort by looking up a hash in the rainbow table before attempting to crack it.

Conclusion

Making it difficult to crack a password is not the goal of password protection; rather, the goal is to make it difficult. For this, having a decent password policy is just as important as having proper password storage: Weak passwords will always be vulnerable to attacks that require testing popular passwords, therefore a purely technical fix may not always be the best option.

PHYSICAL ATTACKS

How Can a Cyberattack Be Physical?

It's no longer true to think of the digital world created by computers as being in a vacuum from the real one. Since computers are tangible objects, physical assaults may be possible against them. Attackers may physically interact with the computer during these attacks, or they may use physical objects like flash drives or smartcards.

Malicious USB Devices

When it comes to physical strikes, smart cards don't get to have all the fun. Malware is frequently distributed through USB drives, also referred to as flash drives. Some malwares can execute immediately automatically and discreetly as soon as the disk is plugged into a computer, even though it is packed in a way that needs a user to manually start it.

However, dangers don't just exist with storage devices: Malicious USB charging stations and cables have been reported in the past. One instance featured a USB charging cord for an electronic cigarette that had a tiny chip buried within the connector that was infected with malware. Without the user's knowledge, it would try to install its payload of malware onto the machine when plugged in.

Furthermore, this is not just a simple trick. Malicious USB devices were initially used to spread Stuxnet, a cyberweapon intended to attack Iranian nuclear enrichment facilities. The centrifuges used to process uranium were subsequently physically destroyed by the virus, demonstrating that cyberattacks can have both physical consequences and physical delivery.

Physical Hacking

An attacker's options expand when they have physical access to a computer. For instance, it might be challenging to steal a password from a secure computer using only digital means. However, the process gets significantly simpler if an attacker can physically install a keylogging device.

By simply unplugging an unencrypted hard drive from one computer and inserting it into another one that can access its information, OS-based security can be disregarded for unencrypted hard drives. It is considerably simpler to search through computer and mobile phone hard drives and locate intriguing or unique files thanks to specialized digital forensics tools. Even though a lot of this software is designed for ethical hackers, it can still be used maliciously in the wrong hands.

Externally accessing a computer's running RAM is one of the trickiest and most sophisticated methods for obtaining information from it. Though challenging and requiring incredibly specialized tools, this is doable.

A network may appear secure behind a firewall, but a hacker with physical access to a network port may connect a microcomputer to the network and then to cellular data. Networks are not immune to physical attacks. The attacker will then be able to target machines on that network without going via the firewall thanks to the microcomputer acting as a backdoor into the system.

(Not So) Smart Cards

Physical hacking attempts frequently target smart cards. It's quite simple to steal information from various types of them because they're frequently utilized for finances or gaining access to secure regions.

One method, known as skimming, involves an attacker using a false card reader to copy or skim data from a card. Credit cards are frequently used for this, but ID cards and other smart card kinds can also use it. In places with little surveillance, like ATMs or petrol pumps, these skimmers are frequently physically affixed to real card readers.

Smart cards can also be "cloned," which goes one step further and involves writing data to a blank card to make a duplicate of an existing smart card. While this information can be obtained via skimming, some smart card kinds are considerably simpler to copy: Smart cards that employ RFID technology can be copied without direct physical contact, and hardware that can fit in a backpack and automatically take the data from any nearby RFID cards is available.

Some penetration testers use this type of cloning as a common tactic. For example, they might browse their phone while sitting on a bench in front of a target building with their backpack next to them. The moment an employee sits down on the same bench, their credit card information is taken.

Conclusion

Security online is crucial. However, in the context of cybersecurity, physical security cannot be disregarded. It's critical to understand how physical attacks can jeopardize assets whether you're managing security for a huge company or yourself.

It is a good idea to enforce security regulations that forbid plugging in unknown USB devices by turning off unused USB ports in machines' BIOS. Keycard cloning can be prevented with the use of RFID blocking wallets and more secure smart cards that do more than just hold data.

SUPPLY-CHAIN ATTACKS

A technique used by attackers to breach one computer and then use it to compromise another is known as pivoting in the field of cybersecurity. The similar concept underlies supply chain attacks, which utilize businesses as opposed to solitary computers.

Assume, for instance, that you are attempting to steal confidential documents from a defense contractor: As required by their contract with the government, the defense contractor most likely has strong security. You do, however, know that the contractor makes use of a piece of commercial software created by a different business that has less robust security. You might introduce malware that gives you a backdoor into the computers it is installed on into the software that the software firm develops by compromising them. You now have access to the defense contractor's machines when they release their upcoming software upgrade.

Practical Examples: Target and SolarWinds

Supply chain assaults have been used in numerous significant breaches. A significant data breach involving Target, a store, happened in 2013, potentially affecting 110 million customers. Using credentials taken from an HVAC company that had done business with Target, the attackers initially got access to Target's network. To monitor Target's HVAC and refrigeration systems, the HVAC company needed access to Target's network, and the attackers were able to utilize this access to penetrate Target's network.

The SolarWinds breach in December 2020 is another, more recent example. A cybersecurity firm called Solarwinds offers software to numerous other businesses, including local and national governments. By hacking into SolarWinds' network, attackers were able to introduce malware into their products, which was then distributed to customers via software updates. The attackers were able to access numerous organizations, including Fortune 500 firms and government agencies, by infiltrating one.

Supply chain attacks highlight how a single weak link in security can have far-reaching implications.

ADVERSARIAL ARTIFICIAL INTELLIGENCE ATTACKS

Artificial Intelligence and Cybersecurity

When attempting to safeguard large or complex environments, artificial intelligence (AI) is a very helpful tool for cybersecurity. AI can assist in identifying unusual or questionable behavior so that analysts can further investigate it.

AI may be used for more sinister goals like detecting security flaws and ways to get beyond protection, though, just like many other cybersecurity tools. Artificial intelligence that is hostile to humans is what this is.

Machine Learning

We can develop algorithms using some types of machine learning that are nearly impossible to program manually. The drawback of these algorithms is that we don't fully comprehend how they decide what to do. It's incredibly challenging to understand the algorithm's "thinking process," even if it correctly determines whether a photo is of a cat or a dog, for instance.

Tainted Training Data

To learn how to work, machine learning algorithms need training data: You'll need a lot of images of both cats and dogs if you want an algorithm that can distinguish between the two. This creates a possibility for bad actors to affect the algorithm: They can alter the behavior of the final algorithm by altering the data used to train it. Tainted training data refers to data that has been maliciously altered.

Sadly, even data that hasn't been deliberately contaminated can be harmful: Unconscious prejudices and accidental errors can quickly skew training data in a negative way. Examples include resume assessment algorithms that discriminate against women or offensive picture recognition algorithms that incorrectly classify people of color. Even if the bias was inadvertent, when biased data is used to train machine learning algorithms, the bias is encoded and continues to be perpetuated.

AI Vs. AI

Even though we might not be able to fully comprehend how machine learning algorithms think, this does not mean that we cannot manipulate them. In fact, other machine learning algorithms are a great resource for creating strategies to deceive machine learning algorithms. We can create data that appears normal to us but tricks the target algorithm into providing responses that make no sense by training one algorithm to deceive another.

When you consider that image recognition algorithms are used for things like autonomous vehicles, spying, and verification, tricking them may seem hilarious. Other kinds of algorithms,

Protecting Your AI

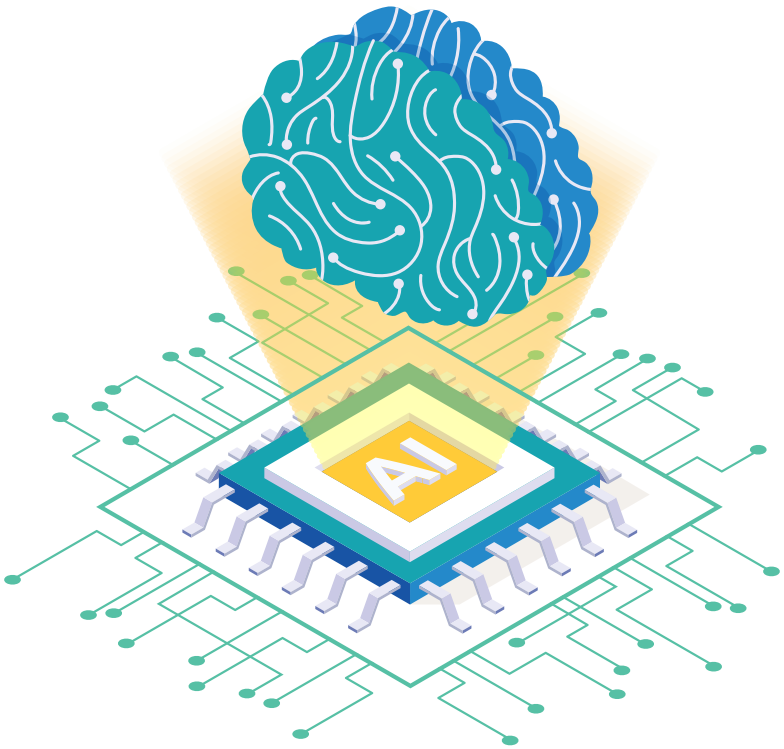
Although there is no infallible method to guard against adversarial AI assaults, there are certain precautions that can be taken:

Maintaining the secrecy of training data makes it more difficult for malicious parties to study the data and can assist prevent unwanted modifications.

Another alternative is to train algorithms to identify and prevent attacks from hostile AI.

Conclusion

Since cybersecurity is a large topic, many other fields—from machine learning to business logistics—will contain some cybersecurity components. It's a good idea to consider the security implications of your projects as you work on them, including potential attack vectors and repercussions. To be "excellent" at cybersecurity, you don't need to know everything about it; all you need to know is enough to engage in critical thought and conduct productive research on the topics you don't fully understand.



WEB APPLICATION ATTACKS

WEB APPLICATION ATTACKS

WEB APPLICATIONS

Web applications themselves can be enticing targets: A web app's vulnerability can be used by an attacker to steal user data, carry out fraud, infect users with malware, and much more.

Web apps can also act as a very noticeable entry point for an attacker trying to get further into the network of an organization: An attacker might target the servers the application runs on instead of the application itself, installing a backdoor to get direct access to them before switching to other computers on the network.

Web applications are sought after because they are visible, easily accessible, and provide a wide range of possible rewards for attackers. Due to the popularity of web applications as targets, groups like the Open Web Application Security Project (OWASP) have been established to inform people about protecting web apps. The OWASP Top Ten is a list of the 10 most serious categories of web application vulnerabilities that OWASP releases.

Most of the threats described in this article can be somewhat avoided by adhering to one straightforward rule: Never trust user input.

Injection Attacks

For some of its functionality, many web apps rely on backend software; some of this infrastructure, such as databases, receives queries based on user input. An injection attack occurs when a malicious query is made by the attacker to deceive the software into performing an action that it shouldn't.

LDAP Injection

A network directory can be accessed via the LDAP protocol. An injection attack against this protocol is called LDAP Injection. Since LDAP frequently deals with credentials like usernames and passwords, a malicious LDAP query can have a significant security impact. Some web applications generate LDAP queries from user input.

SQL Injection

SQL Injection is an injection attack that utilizes SQL, a language used to query databases. Malicious uses for SQL injection attacks include accessing or altering data as well as sending commands to the OS that runs the database. Web applications frequently employ user input to do database queries; a prime example of this is the search function.

Preventing Injection Attacks

Make sure that user input cannot be utilized to influence the software that responds to a query if a web application will be making a query based on user input. While input sanitization, which verifies that queries don't contain control characters used to fool software, is one approach to accomplish this, it is preferable to organize queries in ways that are intrinsically resistant to injection, such as SQL's parameterized queries.

XSS And CSRF

Cross-Site Scripting (XSS) and Cross-Site Request Forgery target the frontend of an online application, whereas injection attacks target the backend.

Directory Traversal

A website's structure may be comparable to the organization of a filesystem on the server it is hosted on. When an attacker has access to restricted areas of the filesystem, this is known as directory traversal, which in file paths means "the folder above this one," is typically used to do this. The simplest method is to just include the.. es in the URL, as in www.example.com/../../...

This can be used to read files that really shouldn't be read, such as the /etc/passwd or /etc/shadow files on Linux, which store user information and password hashes respectively.

Cross-Site Request Forgery

When you are currently authenticated on one website, cross-site request phishing allows that website to send requests to another website. Since this is a little unclear, consider it like this:

Your bank account is open in one tab with you logged in. 2 You open a second tab and browse a malicious website.

The malicious website contains HTML that instructs you to ask your bank's website to do a malicious act.

Your browser delivers the request in accordance with the HTML code.

Your browser sends the request to your bank's website. It fulfills the request's wicked wishes because, as far as it is aware, it is from you.

CSRF tokens, which are randomly generated values created for each session and must be supplied in requests for them to be executed, can protect against this type of vulnerability, which frequently makes use of cookies saved in the browser.

Cross-Site Scripting

When user input on a website is read as JavaScript and executed, this is known as cross-site scripting. XSS vulnerabilities can range in severity from low to extremely high based on the type of XSS and the function of the website.

Reflected XSS is when a website returns the malicious user input immediately, such as a search function displaying your query at the top of the screen. You would need to deceive other people into doing anything, like clicking a URL that had the malicious query included as a parameter, to influence them. Although still potentially harmful, this is not nearly as risky as...

Stored XSS refers to situations where a website stores user input, such as in a forum or post on a social media platform. The input may be significantly more hazardous because it can be seen by many individuals. This kind of XSS has been utilized for everything from sizable-scale credit card data theft to relatively innocent practical jokes like self-sharing social media posts.

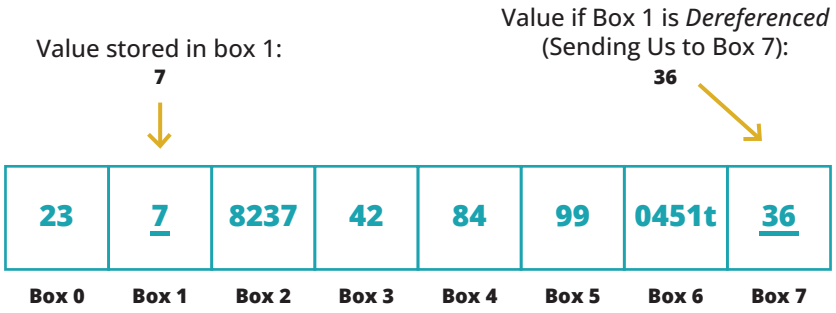
Conclusion

Attackers frequently target web applications, both as standalone targets and as entry points to internal networks. When creating online applications, designers should take vulnerabilities like XSS, CSRF, Directory Traversal, and injection attacks into account. By not trusting user input, several of these vulnerabilities can be mitigated to some extent. This entails validating input, ensuring control characters are handled as text and note code appropriately, and triple-checking the origin of requests.

Pointers

A variable that stores the address of another variable is known as a pointer. The majority of programming languages do not require you to be concerned about pointers, but some do, such as C and C++. Pointers are still used "behind the hood" of the language even if you don't have to worry about them when you're programming.

It is possible to read the value at the address that a pointer points to, a process known as dereference. Consider, for illustration, a row of boxes with the numbers 0 through 15 on them. The number 36 is in box fourteen, which is indicated by a pointer in box one. When you read box one's contents, you get 14, but when you defer to box one, you get 36.



A pointer that points to nothing is known as a null pointer. This is different from a pointer referring to a variable with a value of 0, which is like receiving a piece of paper with the number 0 written on it. A null pointer is equivalent to receiving nothing. Dereference attempts normally result in crashes, but they occasionally allow for arbitrary code execution.

Messing With Memory

Memory Leaks

Because the memory in computers is limited, efficiency is key. Programs will deallocate memory that they no longer need if everything is running smoothly, but mistakes might happen. When memory that has been allocated is not released after it has finished being used, a memory leak occurs. Because of this, a software may gradually start sucking up more and more memory until there is none left, which leads to a crash.

The phenomenon of resource exhaustion, in which a computer runs out of finite resources like memory, disk space, network bandwidth, etc., is shown through memory leaks. Attackers may utilize resource exhaustion in denial-of-service attacks.

DLL Injection

A DLL is a file that other programs can utilize to get instructions. Because they may use DLLs that implement the features they require, programmers no longer must write each of their programs from the ground up.

Regrettably, not all DLLs are reliable. When an attacker can attach a harmful DLL to a trustworthy software, this is known as DLL Injection. The DLL seems to the application to be innocent, but it is acting as an agent for the attackers while disguising itself as a component of the genuine program.

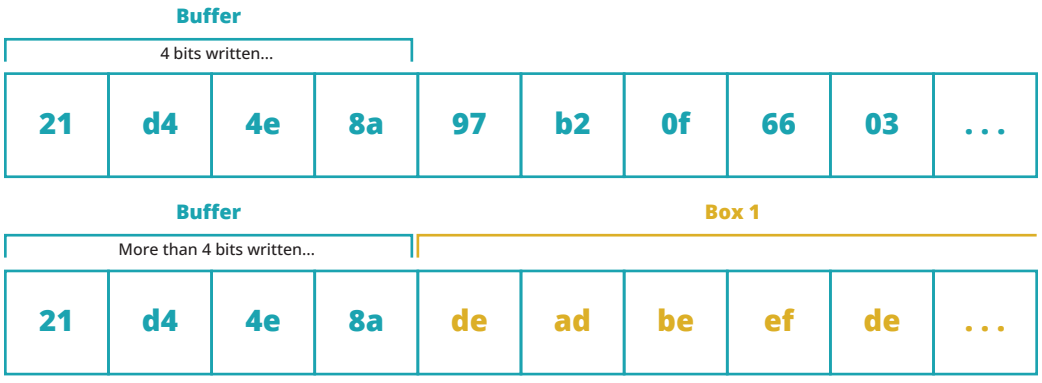
Attackers will try to avoid detection by restructuring their DLLs so that they don't match the antivirus's signatures for known malicious files because antivirus software is aware of malicious DLLs.

Shimming, a technique where a DLL is injected into a program to "translate" out-of-date function calls into ones supported by the current OS, is another way that DLLs are used to preserve compatibility with legacy applications. This is a common technique for introducing harmful DLLs into the software.

Overflow Attacks

It's crucial to be able to manage higher amounts of data because not all of our data neatly fits into a single byte. A buffer, a space of memory allotted by a software to hold data in, is one technique used for this. A software can allocate a 10-byte buffer to hold data if it anticipates receiving 10 bytes of data.

What happens when a buffer is overloaded with data? In the worst-case scenario, a buffer overflow occurs, causing data to keep writing outside of its proper location. Remember that the instructions carried out by the processor are stored in computer memory as well as data. A software may mistakenly overwrite its own code if it writes outside of its buffer, which could result in unexpected behavior.

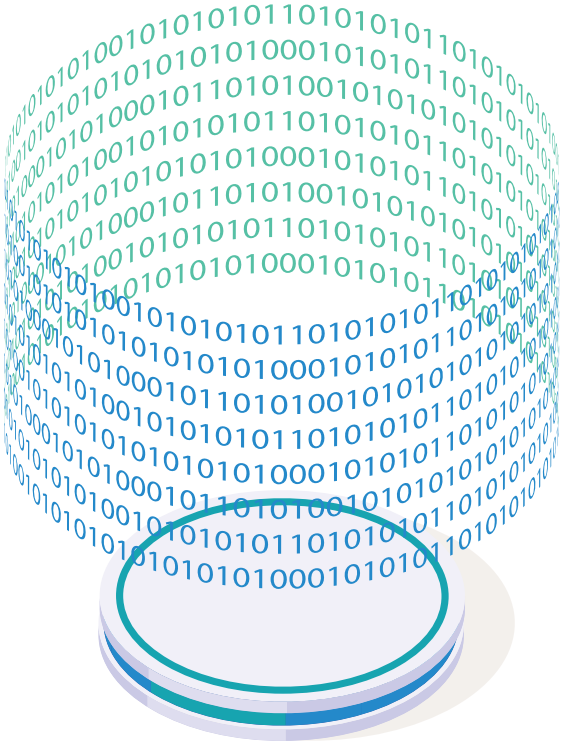


This can be used for a **Buffer Overflow Attack,** in which the attacker deliberately transmits data that overflows a buffer. This enables the attacker to replace the harmful instructions of the program with their own malicious instructions.

An integer overflow attack is a different kind of overflow attack. Integers, unlike buffers, have a fixed size, but this does not preclude overflowing. Depending on how many bytes they use and how they are processed, integers have upper and lower restrictions on the numbers they can contain. If you attempt to store a number that is larger than the maximum, the number will "wrap around" and start over at the lowest value. This can be used by a cunning attacker to make programs behave in an unexpected way.

Conclusion

By making use of flaws that allow for fraudulently created input to alter the program's code, attackers can directly target binary applications. Malicious DLLs can also be used by attackers to inject malicious code into trustworthy apps.



CRYPTOGRAPHIC ATTACKS

Birthday Attacks

Consider that you are searching for a hash collision. It will be more difficult to locate a hash collision if you're looking for one that generates a certain output as opposed to one that generates any output. The method of faking digital signatures known as a "Birthday Attack" takes advantage of this hash collision characteristic. If you have many, slightly different copies of the valid file that generate distinct hashes and you're seeking to locate a malicious file with the same hash, it will be simpler.

Birthday Attacks take their name from the Birthday Paradox, which states that there is a 50% chance that two individuals in a group of 23 will share the same birthday.

Hash Collisions

The outputs of hashing algorithms are fixed in length. For instance, Sha-256 always yields a 256-bit result. However, practically speaking, hashing algorithms typically don't have restrictions on the size of their limitations. This suggests that since there are more inputs than outputs, there must be outputs that can be produced from multiple inputs. A hash collision occurs when a hashing algorithm yields the same result for two different inputs.

In general, it is more difficult to detect hash collisions in hashing algorithms whose output sizes are big.

SSL Stripping

A version of the HTTP protocol protected by SSL is known as HTTPS. In a technique known as SSL Stripping, a cunning attacker can downgrade a connection from HTTPS to insecure HTTP. This enables an attacker to get around the security measures put in place by HTTPS, such as confirming that a website is actually the website it claims to be.

By setting up servers and browsers to exclusively accept HTTPS connections and reject regular HTTP connections, SSL Stripping can be easily avoided. One method servers can use to stop SSL stripping is the HTTP Strict Transport Security (HSTS) header.

Making Use of the Implementation

A door's strength depends entirely on the frame it is mounted in. Even while some forms of encryption are susceptible to mathematical attacks, it is frequently considerably simpler to simply "go past" the cryptographic protection altogether.

Session Replay Attacks

Attacks are known as "Session Replay Attacks" involve the "replay" of stolen data. For instance, even if they didn't have your credentials, if an attacker can intercept a communication where you log into your bank, they could be able to resend the same information and access your bank account.

It is possible to stop session replay attacks by timestamping requests, spotting suspicious data within a request, or making sure that each request is distinctive and only valid once.

Initialization Vectors

The initial state of a cryptographic method is determined by a value known as an Initialization Vector (IV), commonly referred to as a Nonce (Short for "Number Only Used Once"). A well-designed system will make it difficult to predict the IV because a predictable IV indicates that the cryptographic algorithm's output will also be predictable.

Sadly, not every system is well-designed. An initialization vector attack is a sort of attack where an attacker can access apparently protected data by disabling encryption by foretelling the IV used in it. A poorly constructed IV might use a pseudo-random number generator with a flawed design that enables an attacker to anticipate future IVs based on prior IVs, or it might start from the same value each time the machine is turned on.

Pass the Hash Attacks

Passwords that are properly stored are hashed with salt and are incredibly difficult to decrypt. The same procedure is used to verify that a password is typed correctly, and then it is compared to the right password's hash. They must match for the password to be accurate. What happens, though, if the right hash is stolen during an attack?

Given that it would be hashed again and yield a different result, most systems won't be vulnerable if the right hash is merely entered in the password field. Password hashes are nevertheless accepted as authentication credentials by some systems. The Pass the Hash attack, which is frequently used to log onto other computers in a compromised network, is known as that.

Another illustration of this kind of vulnerability would be if a website hashed the user's password client-side instead of using a secure connection for authentication. An attacker might obtain the hash from the unencrypted communication and use it whenever they wanted to log in. Sadly, this is not a made-up instance.

Only if an attacker can obtain the hash are pass the hash assaults feasible. An attacker might be able to steal the hashes of user accounts on a computer if they have local administrative access to it, and they might use those hashes to log into other machines. Systems for password management and vaulting are effective protections in this situation, as is making sure that the least privilege principle is applied to all accounts.

Conclusion

Although cryptography is a strong security technique, it is not a perfect solution. Like every security mechanism, if it is not used correctly or implemented, it can be circumvented and cannot offer full security.

NETWORK ATTACKS

NETWORK ATTACKS

WIRELESS ATTACKS

Wi-Fi Attacks

In essence, Wi-Fi functions as a wireless alternative to an Ethernet connection by connecting endpoints, such as laptops or tablets, to a router through a wireless access point. The majority of routers used in homes are wireless routers, which serve as both a wireless access point and a router.

Wi-Fi Denial of Service Attacks

When an attacker destroys the connection between a victim and a wireless access point, it is known as a Wi-Fi Disassociation attack. In this kind of attack, an attacker impersonating the victim sends the wireless access point a message instructing it to disconnect from the target device. This is a particular kind of DoS attack.

Wireless communication over W-Fi employs radio waves, which are subject to electromagnetic interference. Attacks like this that deliberately exploit this interference to disrupt victims' connections are known as jamming, and they have the potential to entirely stop communication over a channel. It is outrageously unlawful to use electromagnetic interference to intentionally disrupt communications and doing so will probably anger both the FCC (or local equivalent) and nearby amateur radio operators.

Rogue Access Points

A device connecting to a network that has not been authorized by the network administrator is referred to as a rogue access point. This might be anything, from a staff member's personal phone to a malicious backdoor set up by an intruder who physically accessed a network ethernet connection.

Evil Twin Attacks

An Evil Twin Attack is a very straightforward sort of attack in which the attacker sets up a rogue Wi-Fi network that appears to be a trustworthy one. For instance, a hacker might visit a coffee shop that offers free Wi-Fi and then set up their own wireless access point with a name that is quite like the network of the coffee shop. It is highly helpful for Man-in-the-Middle attacks since any unencrypted traffic through the attacker's access point can be intercepted by the attacker. Devices will typically attempt to connect to the stronger access point even if a network with the exact same name as a valid network exists.

Bluesnarfing

Bluetooth is a shortrange wireless communication technology. Wireless accessories like headphones and computer mouse frequently use it to connect to cellphones and computers.

Bluetooth devices must be able to discover each other to function, but this vulnerability can be exploited by attackers via a method called "Bluesnarfing" (Yes, really). To steal data from a targeted device, Bluesnarfing entails searching for Bluetooth devices that are in discovery mode and attempting to exploit OBEX, a mechanism for transmitting data between wireless devices, for security flaws.

A technique is known as "bluejacking" allows attackers to send data to a victim's device. Typically, bluejacking entails sending unwanted text messages to a victim's phone.

NFC and RFID Technologies used for short-range wireless communication include Radio-Frequency Identification (RFID) and Near-Field Communication (NFC).

RFID

RFID employs radio waves to transport data from a "reader," which obtains data from a tag to a "tag," which stores the data. Both passive and active RFID tags are available. Active tags have their own power source, whereas passive tags don't, and instead rely on the reader's wireless power transmission to convey information. Passive tags typically have a reading range of less than a meter, while active tags have a reading range of up to hundreds of meters.

RFID is used to track or identify items like cargo, animals, toll-tag for vehicles, or ID cards.

NFC

Like RFID, NFC is a very short-range wireless communication technology with a range of only a few millimeters. Unlike RFID, NFC may send and receive data in both directions; a phone, for instance, is capable of doing one or the other.

NFC may be used to authenticate users, share information, make payments, and is even integrated into some toys.

Both NFC and RFID technology are susceptible to skimming, in which an attacker gains unauthorized access to the data broadcast over NFC or stored on an RFID tag. Additionally, RFID and NFC might be hampered by electromagnetic interference.

Conclusion

Wireless communication technologies are practical and beneficial, but they also open our communications to new attack vectors. Wireless connections make it easier to eavesdrop and steal data, and many protocols weren't always created with security in mind, so we need to use them with extra caution.

LAYER 2 ATTACKS

Switches
Networking devices called switches forward packages to other nodes in the network. Switches can connect to numerous machines since they have numerous ports. For instance, you might have numerous PCs linked to a switch, which is connected to a router.

ARP Poisoning Attacks

What is ARP?
An IP address's corresponding MAC address can be determined via the Address Resolution Protocol (ARP). What MAC address is associated with IP? is the question that a device broadcasts over its local network segment when it wishes to know the MAC address connected with an IP. Only the device with the IP is expected to respond, broadcasting its MAC address in response. All other devices on the network segment receive this broadcast.

Poisoning ARP
You might be able to spot a problem in how ARP functions because it wasn't developed with security in mind: Only the device with the right IP is expected to respond when a request is made, but there is nothing to stop an attacker from responding instead by giving their own MAC address, which would cause data to be delivered to that MAC address rather than the intended recipient. ARP poisoning is used for a variety of purposes, such as man-in-the-middle attacks and denial of service.

MAC Attacks
MAC addresses have already been mentioned, but now is a good opportunity to examine them in greater detail. Network interfaces are given a hardware address called MAC, which stands for Media Access Control. Since MAC addresses are designed to be unique, no two pieces of hardware should ever have the same MAC address.

Layer 2 protocols use MAC addresses to direct data to the appropriate device.

MAC Spoofing
The initial 24 bits of a MAC address are used to identify the manufacturer, while the final 24 bits operate as a unique ID for that manufacturer. Officially, MAC addresses are divided into two 24-bit portions. However, MAC spoofing—the practice of devices "lying" about their MAC address—is entirely conceivable.

The method of MAC spoofing is NOT complicated, and security precautions like maintaining a deny-list of MAC addresses can be readily gotten through.

MAC Flooding

Switches on a network are the target of the attack known as MAC Flooding. Devices can connect to a switch's various ports, and for data to be sent to its intended location, the switch needs to know which MAC addresses go with which ports. This data is stored in the memory of switches. By bombarding the switch with data that seems to come from many distinct MAC addresses, MAC flooding tries to make the switch run out of space for storing MAC addresses.

Conclusion

Numerous low-level network protocols are weak points for attacks since they were not created with security in mind. These assaults may be intended to disrupt the victim's network or steal information.

DOMAIN NAME SYSTEM ATTACKS

DNS

We can access webpages using URLs thanks to the Domain Name System (DNS). DNS enables computers to identify the IP address that corresponds to a given domain name, enabling us to connect to websites by name rather than having to manually remember a list of IP addresses.

The specifics of how DNS functions may fill numerous articles, but the gist is that DNS operates through a hierarchy-based collection of DNS servers.

Domain Hijacking

If you want a domain name, you must register it with a domain registrar and supply an IP address. After that, the domain registrar will inform the DNS that the domain is now linked to the specified IP address. By using social engineering, hacking, or simply snatching a domain that someone failed to renew, an attacker can acquire control of a domain name without the owner's knowledge.

The domain registrar and occasionally the domain owner are the targets of domain hijacking, not the DNS.

The level of public confidence in a domain is referred to as its domain reputation. Even if the rightful owner can reclaim control, a compromised domain may be used for spam and scams, harming its reputation and leading to the name being blacklisted.

URL Redirection

HTTP requests have a capability called URL Redirection that enables requests to be forwarded to a different page. For instance, you might be led to a login page if you try to see a page of a website that requires you to log in. Although there are many good purposes for this, it can also be employed to deceive victims into going somewhere they didn't intend to. This type of redirection is often only useful if the attacker has access to a website that the target wants to visit, but it can be used in conjunction with other strategies like phishing.

Man in the Middle

The first method an attacker can use to do this is to pose as a local network DNS server. They can act as a DNS server by exploiting ARP poisoning, and they can reply to DNS requests with any IP address they choose.

DNS Poisoning

By interfering with DNS and making domain names point to the incorrect IP address, an attacker can also direct users to websites they didn't plan to visit.

Client Cache Poisoning

The HOSTS file was a method of instructing your computer to associate domain names and IP addresses in the pre-DNS era. This is a file that contains a list of domain names and IP addresses and is kept on your computer. Most operating systems still recognize and respect the HOSTS file, which enables manual association of domain names and IP addresses on computers.

You can probably guess where this is going: If an attacker can add a malicious entry to the HOSTS file on a target computer, that computer will use that entry's IP address to resolve the associated domain name rather than the one provided by DNS. The term "DNS Client Cache Poisoning" describes this.

Server Cache Poisoning

Server cache poisoning targets DNS servers, whereas client cache poisoning targets a client. Not every domain is "known" to every DNS server. DNS servers maintain caches of domains and IP addresses. When they get a request, they first check their cache and, if necessary, request the information from a server further up in the hierarchy.

The DNS server will give a fictitious response to anybody who requests a compromised domain if an attacker can enter malicious data into this cache, at least until the cache is updated. Like the ARP poisoning technique, but on a much bigger scale, this is frequently performed by a mix of Denial-of-Service and impersonation.

Conclusion

Being able to access resources via URLs is incredibly practical for us as humans, but the infrastructure we employ to make this practicality possible could be a target for attackers. Unsuspecting victims could be redirected to a location of the attacker's choice without the victim noticing anything is wrong if the attacker can disrupt domain-name resolution.

ICE

TRUSTED IT PARTNER