SDS Page No.
Date / /

MOD-1

1.1 ✓ Importance & Elements of security,
✓ Phases of an attack
✓ Types of hacker attacks
✓ Hactivism
✓ Ethical hackers

1.2 ✓ Phases of Pen. testing,
✓ Methodologies & Risk.
✓ Pen testing

1.3 ✓ Proper & Ethical disclosure
✓ OWASP Top ten attack.

Page No.
Date

VAPT

Mod-1

Importance of security:

- As we come around a digital world, every process involves or is under process of being involved with digitization.
- Data is found everywhere, networks are being established.

→ As our dependence on the technology increases, there is also a threat to data & thus our dependence on ~~cyberspace~~ Cyberspace & a need to keep our data safe & secure

→ Data privacy, network privacy, network integrity & functioning that keeps our lives streamlined need cybersec. to run seamlessly.

Elements of security:

We want to protect our computer sys. from any harm.

→ Security Researchers & analysts have come up with some unique concepts to keep system safe & secure.

→ If anyone of the constraints / element is compromised, it possess potential risk to the system:

a) Availability → whether data or resource is available as and when required / requested by user.

→ cybercriminal seize those data, thus req. to access that data gets rejected. (server downtime).

b) Integrity → technique to ensure that all the data i.e. available to be accessed in real-time is legitimate. & protected ~~under~~ from unlawful user modificatn. via checksums, data comparisons, etc. the data integrity is verified.

c) Authenticity → can be defined as process of ensuring & confirming the identity of user is legitimate. Authenticatn takes place when user tries to access, any data or information

d) confidentiality →

SDS | Page No.
Date / /

- Confidentiality → permitting approved user of accessing sensitive data. Confidentiality can be achieved using role based security techniques to ensure user & viewers authorzatn. & access control on any data.

- Non-repudiation → method of assurance, that the message was transmitted among the 2 or 1 more users via digital signature or through encryption is accurate. Mentioning the authenticity of message / doc.

- utility → used for any purpose, the data is accessed & then used by user. Not entirely the type of element for security, but if any resources utility becomes vague, then it is of no use.
  cryptography is used to preserve the efficiency of any resources ~~used~~ sent over internet. ~~Most~~ maintainging the authenticity of data (not altered), the utility wont prevail.

- Accountability → Identificatn & authentcatn among the system, user and application, thus all these must be noted down.

- Reusability → A user may not reuse / manipulate the data that is currently accessed by some other entity to prevent security threat.
  Not all resources are available to everyone at all times.

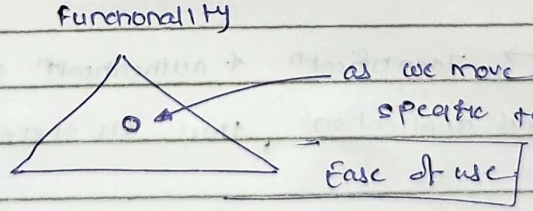
Security Attacks (types & examples)

- Threat to confidentiality → snooping, Traffic analysis.
  - ↓ unauth. access to data
  - ↳ keeping an eye on packets sent over networks.

- Threat to integrity →
  - Modification, (data manipulation)
  - Masquerading,
  - Replaying, (reuse of packets during diff times).
  - Repudiation (rejecting the authenticity of msg).

- Threat to availability → Denial of service (DOS)
  - ↓ making the server busy by loading it with unwanted / false requests.

Security, Functionality & Ease of use triangle



as we move towards any specific target, we lose other two.

Phases of an attack

1) Reconnaissance → Primary stage of attack, where we gather the information about the target. Any data such as IP, DNS, s/w etc. Then trying to identify one vulnerable target & gain entry to system via them.

Basic 2 types → Active & passive reconnaissance

2) ~~Weaponisation → To info obtained via reconnaissance, we need a method to deliver that. & Generally the network is compromised by~~

2) Weaponisation → To weaponise the info obtained via reconn. That means, need to decide the s/w sytem to break down the target system and also exploit the network.
→ To compromise the network done by unpatched security s/w on connected device is exploited / attacked.

3) Delivery → at this point attacker has his target, info on target & a method choosed to attack the target to gain control.

4) Exploitation → attacker makes the attack, breaching the target system via choosen delivery method.
Gaining much more info on how traffic flows, connected system & user flows: leading to more exploitatn.

5) Installation → As attacker wants to continue the access over system, attacker creates a backdoor, disabling the firewalls & creating the accounts with admin access.
Thus, can install any malicious s/w & can trigger remote desktop, to continue the access.

6) Command & control → Now when all things are set, required tools are placed to trigger command & control, that attacker getting effective control to obtain curr info.

4) Actions & objectives → the final stage could involve stealing, disrupting, demanding ransome, & more over exploiting the data.

* phases of hacking

a) Reconnaissance → gather info. (8ts).

b) Passive reconnaisance → collects info about target system, network or org. using open source i.e. available info. about DNS; Domain names, emails etc.

c) active Reconnaissance → active reconn meaning direct interaction with target system, detecting open ports, hosts, routers, etc.

d) Scanning → attacker uses details gathered during reconn. to identify vulnerabilities, More indepth gathering of data on victim.
logical extension of active reconn.

e) Gaining access → If vulnerability is found, attacker, exploiting that tries to access the system. ; network, app.
An unauth. access to the systems OS is done.
Attacker can now, exhaust systems resources, or may perform many attacks like (DOS, session hijacking, etc)
A hackers chances of gaining access to system are based on his skills.

f) Maintaining Access → Once the access is gained, attackers accounts to stay in the system. to fulfill the goal of attack. Once the attacker is into system, they can use both system and its resources to exploit others. They may install various viruses to gain the access back to the system (eg. Trojan horse to enter app.). Though various systems now use honey pots to trap hackers.

g) covering tracks → Attacker wants to destroy all his traces, for that, trojans are generally used to erase attackers activities. once trojans are placed into system, attacker is likely to gain all access of the system.

Types of Hacker attacks ⇒

There are many ways an ~~bad~~ hacker can get into the system, exploit a systems weakness & vulnerabilities.

1) o.s. attacks → as nowadays O's's comes with various features, some are more prone to vulnerabilities for attackers. Attackers constantly look for OS vulnerabilities that will allow them to get access to the system. By default most of the O.S. installation program, install a large no. of open ports, leading attackers for vulnerabilities.
→ It is necessary to get patches to protect the system
OS vulnerabilities include → Buffer overflow vulnerabilities.
→ Bugs in OS
→ unpatched OS
→ Breaking file system security. etc.

2) Misconfiguration attacks →

- Poorly configured security controls might allow attackers to gain unauth. control to system & resources.
- Misconfig. vulnerabilities affect webservers, databases, network or framework leading to gain unauth access.

→ config. must be made default before deployment of any app. or s/w.

3) Applicatn level attacks →

- Due to time constraints, the testing of any s/w may be not done properly leaving behind undiscovered security holes.
- As increase in complexity means more no. of vulnerabilities, making it easy for attackers to ~~acces~~ gain unauth access. leading to issues like → cross-site scripting, session hijacking, DOS, SQL injection, etc.

* Session Hijacking → exploit the info. in session, if the code implements a cookie less authenticatn.
→ The authenticatn token is captured & replaced with a known token (to attacker) thus gaining access & also steal of info.

* D.O.S → restrict the use of resources of the system, either by flooding the server with false requests. or by any traffic overflow method.

4) Shrink - wrap codes

- s/w developer often use libraries & codes from various other resources in their code to reduce the time
- If the code / library that is copied from somewhere

may contain any vulnerability that may provoke a security threat.

Hactivism

* Hacking → Exploiting system vulnerabilities to gain unauth access over the system and its resources. Can modify the system / application feature to achieve their goal.

* Hacker → person who breaks into the system w/o authorization to destroy, steal or manipulate data.

Classes → Black hats (criminals), white hats (pen tester), Gray hats (mix), suicide hackers, script kiddies, cyber terrorist (religious), state sponsored (agenda), Hactivist.

→ Hactivism is when hacker breaks into govt. or corporate computers as an act of protest.

→ Hactivists use hacking to increase awareness of their social or political agendas.

→ [eg by disabling (taking down) websites to prove their agenda.

→ common hactivist target includes govt. agencies, mncs, or other entity that they percieve as threat.

⇒ 'Anonymous' is a world wide famous hactivist group.

Ethical hacker →

• Ethical hacker also follows the same processes as that of a malicious hacker to gain access over a system, resources w/o authorised access.

- Yet this is done, in order to assist the organisation (by themselves) in order to test their networks for possibles security holes or vulnerabilities.

- White hats ~~are~~ is the term refered to them (pen tester), they help org. in enhancing cybersec.
  Done with the permission of network, system founer to audit the system for vulnerability.
  → generally a person, who thinks like a cracker & tries to find out new possible vulnerabilities.

- their task is to evaluate the system security, update admin regarding all discovered vulnerabilities, & finding procedures (patches) to fix em.

Task done by ethical hacker

a) uncover vulnerabilities in the systems
b) analyse & strengthen systems security policies, network infrastructure;
c) help safeguard customers data

- Tech as well as non-tech skills are required.
  ↳ requires indepth knowledge about the OS env., networking concepts, knowledge to launch tech. attacks
  adapt quickly, work ethics, uptodate with local stds.

1.2

Penetration testing ~~stages~~ :

- A security exercise, done by cyber expert where they attempt to find & exploit vulnerability.
  Main purpose is to identify weak spots (security holes, vulnerabilities) that attacker could take adv. of.
- Performed by 'ethical hackers'.

Pen testing stages (similar to phases of attack)

a) Planning & reconnaissance

- Define the reason (goals & scope) of test, including system to be addressed & testing methods to be used.
- Collect info about the system. for easy penetration.

b) Scanning

- Deep dive of research or info collection. to understand the target system indepth.

c) Gaining access

- Based on information, we get many backdoors, open ports to uncover target's vulnerabilities.
  catching the vulnerabilities & gaining unauth access to system.

d) Maintaining access

- Goal now after breaching is to stay in the system. For that generally trojans are used. The hacker needs to stay in the exploited system - long ~~ang~~ enough for a bad actor to gain in-depth access. Generally idea is to stay in system for long.

and also, corrupting equally imp. as leave any of system.

up to hacker won't the traces in

e) Analysis

- A detailed report is made, based on pen-testing, mentioning the vulnerabilities that were exploited, sensitive data that was accessed & amt. of time pen tester was in system.

This info is analysed to help configure org's their security & patch vulnerabilities.

Pen Testing Methodologies

a) Internal testing

→ A (tester) with access to an application behind its firewall simulates an attack by a malicious insider.

→ generally simulating as a rogue employee but considered as employees whose cred's are stolen.

b) External testing

→ External pen (tester) target the assets of company that are available on internet eg: webapp, website. to gain & extract valuable data.

c) Blind Testing

→ A (tester) only ~~agency~~ is given the name of the org. to be targeted. And is set free, thus security personal can check by all means how actual app. assault would take place.

d) Double - blind testing

→ (Security personal) has no prior knowledge of sim. attack. Just like real world, they coont have any time for defence.

(org → organisation)

e) Targeted testing

→ Both tester & security personnel work together to keep eachother updated of their movements. Providing security team with a real time feedback. from hacker pov.

Risks in Pentesting

a) System outages →

- During pen testing, the tester ~~is~~ is going to exploit the vulnerabilities in the system, but while doing so, they might break into something important by accident, leading to system outages.
- Rashness ⇒ not done on purpose but due to inexperience or inattentiveness, tester may misuse tools that may lead to system outages.
- unexpected circumstances may occur, due to any unforseen reasons like misconfigurahn, etc.

b) Complacency during Pen testing ⇒

- Org may fail to identify an attack on system while conducting pen testing.
  (Failing to recognise an actual attack during pentesting)
- To solve this, there should be proper communication betn tester & security officer. to check on IP addresses in white list.

c) Decrease in Productivity

- certain attacks while pen-testing like DOS, MIM, may restrict the employees to the internet.
- or troubleshooting during pentesting may consume time.

- Org must make employee already aware of pen testing happening to help ~~test attack~~ to with efficient P.T.

c) False Negatives ⇒

- Vulnerabilities, that are not found by Pen-tester.
- Thus org's, defence against cyber attacks is just not limited to pen testing but also regular patching is required to make sure all sec. efforts are best practices.

c) Enethical hackers ⇒

- Risk that the pen-tester is not honest. & has unethical motives ~~too~~ towards system / data.
- May be an hactivist, or doing for money (ransome)

1.3

Proper & Ethical Disclosure

- Method where an ethical hacker, reports any security flaw or issue found during pen-testing, to the organisation.
- Main purpose of this is to inform the org. & the customer about the potential risk, thus further actions can be taken.

→ we need to inform the org. about the vulnerability.
→ the info. regarding this is to be shared in confidential manner

(x) generally we have 8 types of disclosures

a) Private disclosure → reports is provided to org. privately & depends on org. whether to disclose about issue or not. Issues is that, sometimes org do not pay attention to this, may lead to data gayab!

b) Public disclosure → (Full disclosure), researches report the flaw publically, even w/o the fix been attempted.
Putting pressure on org. to fix the issue.
But this comes with many risks, as vulnerability is disclosed.

c) Responsible disclosure → Researcher discloses vulnerability publically, but only after the org has a fix to the vulnerability.
Based on a std. timeframe, the vulnerability is disclosed.

OWASP (Open web-application security project)
owasp.org
→ a non-profit org focused on security of slw. They include various tool kits, local chapters & conferences.

* Owasp top 10 → is a list of top 10 most common web-app. security risk

① ~~A1:2021~~ Broken Access Control (BAC) ~~4% of app. were tested for some form of BAC. The 34 common weakness enumerations~~

1) A1) Broken Access Control (BAC).
→ If auth. & access restrictions are not properly implemented, it is easy for attacker to get it with BAC flaws, unauth user may have access.
→ access control policies, restrict user to their intended data permissions.

2) A2) Cryptographic failures
→ common errors like hardcoded passwords, outdated crypto algos, may result in exposure to sensitive data.

3) A3) Injection

→ when application accepts untrusted data, this is a common vuln. They include SQL injectn, OS commands injectn etc.

→ attacker may i/p code, thus getting into dB's & potentially (eventually) gaining system access.

can be revealed via remediation techniques.

4) A4) Insecure Design

→ Vulnerabilities that exist due to lack of security implementation, at time of development.

→ This tells that, no proper security model was used in designing app.

5) A5) Security misconfig

→ security controls (eg: access) may be misconfigured or left insecure, putting the system and data at risk

→ revealing major data than expected.

reducing misconf. risk org. should routinely harden deployed app. & infrastructure. config.

A6) Vulnerable & outdated components

→ A large no. of 3rd pary libraries are used now, & many app may have security codes from lib that may have vulnerability.

→ These libraries are patched, but what module is used in our app, needs to be considered and patched.

→ component based vulnerability is seen when any component becomes out-dated i.e. unavailable. (api).

A7) Identificatn & authenticatn failures

→ Identifying & authorsing users and non-human clients is a fundamental security practice.

→ Way of identifying users is critically vulnerable.

→ default login creds, cryptography failures, etc.

A8) Software & data integrity failures

→ Tools used in building, manging & deploying s/w are common vectors of attack.

→ using critical data / apps w/o verificaton of their identity falls under this category.

A9) Security logging & monitoring failures

→ no direct vulnerability can arise due to this, but logging & monitoring are quite critical & their absence or failures can directly impact visibility towards app.

→ having a functional logging & monitoring is essential ~~to~~ in both breaching, hopefully limiting the damage. & knowing the scope of breach

A10) Server Side Request Forgery (SSRF)

→ Now many web-apps commonly fetch additional ~~details~~. contents or data from remote resource.

→ If attacker can influence the destination source, & app. does not validate the URL, then crafted req. may be sent to target destinatn.