**DDOS**

**Ping Flood**

Exploit ICMP

Attack principle: The attacker sends many ICMP echo request packets to the targeted server using multiple devices. The targeted server then sends an ICMP echo reply packet to each requesting device’s IP address as a response.

Solution: disable the ICMP functionality of the target device

**OSI 5 Layer Model**

link/IP layer: send too much traffic for switches/routers to handle

transport layer: require servers to maintain large number of concurrent connections or state

application layer: require servers to perform expensive queries or cryptographic operations

**TCP SYN Flood**

Attack principle: SYN packets with random source IP addresses; Fill up backlog queue on server; No further connections possible

Solution: increase backlog queue size; decrease timeout

**SYN Cookies**

avoid state storage on server until 3-way handshake completes

server sends necessary states to client along with SYN-ACK; client sends these states back to server along with ACK;

T: 5-bit timestamptime() logically right-shifted 6 positions; M: 3-bit MSS

L = MACkey(SAddr, SPort, DAddr, DPort, SNC, T)

**Smurf Attack**

Attack with an ICMP Echo Request with spoofed source IP address of the targeted server and destination IP address of an IP broadcast address

Solution: disable IP broadcast addresses on router and firewall, or reject external packets to brdct addr

**DNS Amplification Attack**

Attack with an ANY-type DNS query with spoofed source IP address of the targeted server

Solution: reduce the number of open resolvers; source IP verification

**NTP Amplification Attack**

Exploit monlist command that triggers a response with the last 600 source IP addresses of requests made to the NTP server

Solution: reduce the number of NTP servers that support monlist; source IP verification

**SSL/TLS Flood**

Exploit SSL/TLS handshake requests to drain server resources

RSA-enc speed = 10x RSA-dec speed

**HTTP Flood**

Complete real TCP connection; Complete TLS Handshake; GET/POST large image or other content

**Fragmented HTTP Flood**

Split HTTP packets into tiny fragments

Send fragments to the target as slowly as it allows before it times out

keep a resource-consuming connection active for a long time

**DDoS defenses**

enrich server with more resources; leverage the sources of others; detect and filter attack traffic with spoofed IP addresses

**Ingress Filtering**

ISP only forwards packets with legitimate source IP

Implementation challenges: All ISPs need to do this — requires global coordination

**Secure Routing**

**Delivery Scheme:** Unicast, broadcast, multicast, anycast, geocast

**routing attacks**

distance-vector: announce 0 distance to all other nodes

link-state: drop links; claim direct link to other routers

BGP: announce arbitrary prefix; alter paths

**Prefix Hijacking**

AS claims ownership of some IP prefixes, but it doesn't

AS claims to have a smaller range of IP prefixes than the autonomous system that actually declares to have an IP prefix

**Path Tampering**

AS claims it can deliver data to the hijacked autonomous system via a shorter path than is known

Remove/Add ASes in the AS path

**RPKI**

certified mapping from ASes to public keys and IP prefixes

Cannot avoid path tampering

**S-BGP**

Each AS on the path cryptographically signs its announcement

Deployment challenges: Complete, accurate registries; Public key infrastructure; Cryptographic operations; Need to perform operations quickly; Difficulty of incremental deployment

**Anonymous Communication**

**Overlay Network**

Handle routing at the application layer

Tunnel messages inside other messages

**Anonymizing Proxy**

intermediary between sender & receiver

Sender relays all traffic through proxy

Encrypt destination and payload

Asymmetric technique: receiver not involved anonymity

k: shared key of sender and proxy



Advantages: Easy to configure; Require no active participation of receiver, which need not be aware of anonymity service; Have been widely deployed on Internet

Disadvantages:Require trusted third party proxy may release logs, or sell them, or blackmail sender; Anonymity largely depends on the (likely unknown) location of attacker

**Crowds Algorithm**

Relay message to random jondo; With probability p, jondo forwards message to another jondo; With probability 1-p, jondo delivers message to its intended destination

**onion routing**

Connect to Tor entry; Randomly select a series of Tors; Relay messages across them; Tor exit relays messages to destination

Reply traffic from destination traverses the reverse path

Maintains a bidirectional persistent multi-hop path between source and destination

Layered Encryption: {{{{msg}D,D}C,C}B,B}A

Leaked routing info: neighborship only

**De-Anonymization**

**Tor Traffic Correlation**

Passive monitoring

Active attraction: deploy a Tor router; attract Tor traffic; perform traffic analysis and correlation;

**Path Selection Attack**

weight nodes by selfreported bandwidth; select each node using weighted probability distribution;

Attack: malicious relay reports very high bw to increase selection probability; if it controls the first hop, de- sender; if it controls the last hop, de- receiver;

**Counting Attack**

Correlate incoming and outgoing flows by counting the number of packets

**Low Latency Attack**

Tor router assigns each anonymous circuit its own queue

Dequeue one packet from each queue in round-robin fashion

**Cross Site Attack**

Crawling: Deploy Tor routers; Access darknet; Crawl transaction information; Extract Bitcoin accounts of interest

Correlation: Search the accounts on public websites

**Web Security**

**Web Security Goals:** Integrity, Confidentiality, Privacy, Availability

**Same-Origin Policy**

Each site in the browser is isolated from all others

Multiple pages from the same site are not isolated

Origin = Protocol + Hostname + Port

One origin should not be able to access the resources of another origin

**CSRF**

Exploit cookies that a web server uses to identify a user within a connection session

It is possible for third-party websites to forge requests that are exactly the same as the same-site requests

The server cannot distinguish between the same-site and cross-site requests

**CSRF Defenses**

Referer Validation

CSRF Token: a unique, secret, unpredictable value generated by the server-side app and transmitted to the client

included in a subsequent HTTP request made by the client

the server-side application validates that the request includes the expected token and rejects the request if the token is missing or invalid

**XSS Attack**

Stored XSS: attacker leaves JS lying around on web service for victim to load

Reflected XSS: attacker gets user to click on specially-crafted URL with script in it, web service reflects script back

**XSS Defenses**

Input Validation: check that inputs are of expected form (whitelisting instead of blacklisting);

Output Escaping: escape dynamic data before inserting it into HTML

CSP: Content-Security-Policy HTTP header allows the response to specify white-list, instructs the browser to only execute or render resources from those sources

**Email Security**

**Email Security Threats**

Authenticity-related: could result in unauthorized access to an email system

Integrity-related: could result in unauthorized modification of email content

Confidentiality-related: could result in unauthorized disclosure of sensitive information

Availability-related: could prevent end users from being able to send or receive email

**S/MIME**

Authentication

1. the sender creates a message

2. use SHA-256 to generate a 256-bit message digest

3. encrypt the message digest with RSA using the sender’s private key; append the result as well as the signer’s identity to the message

4. the receiver uses RSA with the sender’s public key to decrypt, recover, and verify the message digest

Confidentiality

1.the sender creates a message and a random 128-bit number as a content-encryption key for this message only

2. encrypt the message using the content-encryption key

3. encrypt the content-encryption key with RSA using the receiver’s public key and append it to the message

4.The receiver uses RSA with its private key to decrypt and recover the content-encryption key

5. use the content-encryption key to decrypt the message

**PGP**

Differences from S/MIME:

Key Certification: S/MIME uses X.509 certificates issued by CA or delegated authorities; OpenPGP allows users to generate their own OpenPGP public and private keys, and then solicit signatures for their public keys from known individuals or organizations

Key Distribution: OpenPGP does not include the sender’s public key with each message; recipient needs to separately obtain that from TLS-protected websites or OpenPGP public key servers; no vetting of OpenPGP keys, users decide whether to trust on their own

**DANE**

allow X.509 certificates to be bound to DNS names using DNSSEC

**TLSA Record**

A new DNS record type defined by DANE

Used for a secure method of authenticating SSL/TLS certificates

Specify constraints on which CA can vouch for a certificate, or which specific PKIX [Public Key Infrastructure (X.509)] end-entity certificate is valid

Specify that a service certificate or a CA can be directly authenticated in the DNS itself

**DANE for SMTP**

Targeted vulnerabilities: attackers can strip away the TLS capability advertisement and downgrade the connection to not use TLS

TLS connections are often unauthenticated

A domain can use the presence of TLSA as an indicator that encryption must be performed, thus preventing malicious downgrade

A domain can authenticate the certificate used in the TLS connection setup using a DNSSEC-signed TLSA

**DANE for S/MIME**

Introduce a SMIMEA DNS record to associate certificates with DNS domain names

Help MUAs to deal with domain names as specified in email addresses in the message body (rather than domain names specified in the outer SMTP envelope – purpose of TLSA)

**SPF**

ADMDs (Administrative Management Domains) publish SPF records in DNS specifying which hosts/IP-addresses are permitted to use their names; receivers use the published SPF records to test the authorization of sending Mail Transfer Agents (MTAs) using a given “HELO” or “MAIL FROM” identity during a mail transaction

**DKIM**

sign email message by a private key of the administrative domain from which the email originates; at the receiving end, the MDA can access the corresponding public key via a DNS and verify the signature, thus authenticating that the message comes from the claimed administrative domain

Difference from S/MIME and PGP: S/MIME and PGP use the sender’s private key to sign the content of the message; DKIM uses the private key of the domain where the sender locates

**Attack Traceback**

**IP Traceback**

router adds its own IP address to packet

victim reads path from packet

Assumptions: trusted routers; sufficient packets to track; stable route from attacker to victim

Limitations: requires space in packet; path can be long; no extra fields in current IP format (changes to packet format too much to expect)

Sample and Merge: store one link in each packet; router probabilistically stores own address; fixed space regardless of path length

**ICMP Traceback**

iTrace

Each router samples one of packets it is forwarding and copies the contents and adjacent routers’ info into an ICMP traceback message

Router uses HMAC and X.509 digital certificate for authenticating traceback messages

Router sends ICMP traceback messages to the destination

Require all the routers transmitting attack traffic be enabled with iTrace to construct an entire attack path

yet ICMP packets are usually filtered… because of ICMP Ping Flood Attack…

yet not all packets are sampled on every hop

**Link Testing**

Traceback from the router closest to the victim

Determine the upstream link that is used to carry out the attack traffic

Recursively apply the previous technique until the attack source is reached

**Input Debugging**

Find attack signature, the common feature contained in all attack packets

Communicate the attack signature to the upstream router, which then filters attack packets and determines the port of entry

Recursively apply the previous technique on the upstream routers until reaching the attack source

A considerable management overhead at the ISP level to communicate and coordinate the traceback

**Controlled Flooding**

Need collaborative hosts

Force the hosts to flood the links to upstream routers

Since buffer on victim is shared by all incoming links, flooding the link carrying out attack leads to drops of attack packets

Recursively apply the previous technique on the upstream routers until reaching the attack source

Require an accurate topology map

High overhead given multiple attacking sources (e.g., DDoS)

**Network Protection**

**Firewall**

Form a barrier through which the traffic going in each direction must pass

Use firewall security policy to dictate which traffic is authorized to pass in each direction

All traffic from inside to outside, and vice versa, must pass through the firewall.

Only authorized traffic, as defined by the local security policy, will be allowed to pass.

The firewall itself is immune to penetration.

**IDS**

Detect unusual patterns of activity or patterns of activity that are known to correlate with intrusions

Provide early warning of an intrusion so that defensive action can be taken

**IPS**

an extension of IDS to attempt to block or prevent detected malicious activity

Anomaly detection: to identify behavior different from legitimate users

Signature/heuristic detection: to identify malicious behavior

**Honeypot**

Decoy systems designed to lure a potential attacker away from critical systems

Collect information about the attacker’s activity

Encourage the attacker to stay on the system long enough for administrators to respond

**Honeywords**

Associate false passwords (honeywords) with each user’s account

Attacker that steals (hashed) password file cannot distinguish from passwords from honeywords

Attempted login using a honeyword sets off an alarm

**Load Balancing**

Distribute network traffic across multiple servers; Mitigate single point of failure

Least Connection Method, Least Response Time Method, Round Robin Method, IP Hash

**Traffic Scrubbing**

Use a data cleansing service that analyzes traffic and filter maliciou traffic

Such service provider should be equipped with sufficient resources to sustain high volumetric floods

Once an attack is detected, redirect traffic to scrubbing service

Analyze and filter malicious traffic

Deliver clean traffic to network/user

**User Authentication**

Identification Step: present an identifier to the security system

Verification Step: present or generate authenticaton information that corroborates the binding between the entity and identifier

**Salt Purpose**

Prevent duplicate passwords from being visible in the password file

Greatly increase the difficulty of offline dictionary attacks

Greatly increase the difficulty of finding out whether a person has used the same password on two or more systems

**Token:** Objects that a user possesses for the purpose of user authentication

**Biometric:** Authenticate a user based on the unique physical characteristic

**Access Control**

Implement a security policy that specifies who or what may have access to each specific system resource and the type of access that is permitted in each instance

**DAC**

Discretionary Access Control

Access Matrix; Access Control List; Capability List

**RBAC**

Role-Based Access Control

Assign users with different roles according to their responsibilities

Check the roles that users assume in a system rather than the user’s identity

**ABAC**

Attribute-Based Access Control

Define authorizations that express conditions on properties of both the resource and the subject