DDOS

Ping Flood(symmetric) Exploit **ICMP**

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

Solution: disable ICMP functionality of target device

OSI 5 Layer Model

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

transport: require server to keep many concurrent connection/state

application: require server do great query/cryptographic operation

TCP SYN Flood(symmetric)

Attack: SYN packet 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 timestamp logically right-shifted 6 positions; M: 3-bit MSS

L = MAC_{key}(SAddr, SPort, DAddr, DPort, SN_C, T)

Smurf Attack(Asymmetric)

Forward **single ICMP Echo** Request to any other hosts in same network; Each host responds with an ICMP Echo Reply

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

DNS Amplification Attack(Asymmetric)

Attack with an ANY-type DNS query to DNS resolver with spoofed src IP of targeted server; DNS resolver then send EDNS to target server

Solution: reduce number of open resolvers; source IP verification

NTP Amplification Attack(Asymmetric)

use botnet to send UDP by spoofed IP(victim) to NTP server(has monlist).Each UDP req server by monlist, send large rsp to victim

Sol: reduce #NTP server(support monlist); src IP verification

Memcached attack preload large data to Memcached server; spoof request to preloaded data from target by GET**; SSDP attack**

SSL/TLS Flood(Asymmetric, computation)

Exploit SSL/TLS handshake request to drain server. enc faster than dec

HTTP Flood

Complete real TCP connection&TLS Handshake; **GET/POST large** image/other content Sol: block/rate limit attacking source

Fragmented HTTP Flood

Split HTTP pkt into tiny fragments; Send them to target slowly as allow before time out; keep resource-consum connection active for long time

Tail Attack(Asymmetric, from weakest link) Saturate weakest link w/ low-rate traffic

SDN CrossPath Attack Disrupt SDN control channel by shared link block control msgs with attacking traffic

DDoS defense(attack harder 1~3 ; attacker consume more 4~5)

Ingress Filtering=ISP only forward pkts with legitimate source IP Implement challenge: global coordination(All ISP need to do)

Traceback by edge sampling(p: write R to start addr,0 to dis field;1-p: write R to end addr if dis==0,dis++) basis(many pkt;stable path; trusted router) use path validation to check malicious router

Alibi Routing(verify pkt **NOT** transmit by specific AS) proof waypoint

Client Puzzles(let C do some consuming computation) **CAPTCHA**

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 Cannot avoid path tampering

certified mapping from ASes to public keys and IP prefixes

S-BGP

Each AS on the path cryptographically signs its announcement. validate AS path indicates the order ASes were traversed, No intermediate ASes were added or removed

Address attestation: Claim the right to originate a prefix; Signed and distributed out-of-band;Checked through delegation chain from ICANN **Route attestation:**Distributed as an attribute in BGP update msg; Signed by each AS as route traverses network;Signature signs previously attached signatures

Deployment challenge: Complete&accurate registries(prefix ownership); Public key infrastructure(know public key for any AS); Cryptographic operation(digital signature on BGP msg); Perform operation quickly(avoid delay response to routing change); Difficulty of incremental deployment(Hard to have "flag day" to deploy S-BGP)

Anonymous Communication

Overlay Network

Handle routing at **application** layer; Tunnel msgs inside other msgs

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; widely deployed on Internet

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

Crowds Algorithm (proxy++ to evade untrusted proxy)

Relay msg to random jondo; probability p, jondo forward msg to another jondo; probability 1-p, jondo delivers msg to its intended destination

onion routing(source based routing)

Get list of node from directory node, random select series of Tors; 2. Get PK from directory, use it to negotiate with A, A negotiate with B, B negotiate with C until whole chain established 3. Layered Encryption: $\{\{\{\{msg\}_D,D\}_C,C\}_B,B\}_A$;4. Reply traffic from dst traverses reverse path; Maintain bidirectional multi-hop path between src&dst

Leaked routing info: neighborship only **(POF based routing** may leak port seq(only leak to neighbor keep anonymity))

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 node by self-reported 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 number of packets

Low Latency Attack JY

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 Goals: Integrity, Confidentiality, Privacy, Availability

SQL Injection(server side), others are client side

Prepared statement seperate data&code.DB parse/compile on statement; later bind data to prepared statement(excute)

Same-Origin Policy(enforced by browser)

Each **site** in browser isolated from others; Multiple page from same site not isolated

Origin = Protocol(http) + Hostname(coolsite.com) + Port(81)

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

CSRF(Cross-Site Request Forgery)

Exploit cookie that web server uses to identify user within a connection session(**secure cookie only sent by https**)

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 same-site and cross-site requests

CSRF Defenses

Referer Validation(add 'referrer' to header of packet)

CSRF Token: a unique, secret, unpredictable value generated by server-side and transmitted to client; token is included in subsequent HTTP request made by client; server-side app validates request includes expected token and rejects request if token missing/invalid

XSS(Cross-Site Scripting) Attack

Stored XSS: attacker leaves JS lying on web service for victim to load; Attack happens **within** the same origin

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

XSS Defense

Input Validation: check input is of expected form (whitelisting instead of blacklisting);

Output Escaping: escape dynamic data before insert it into HTML

CSP(Content-Security-Policy)HTTP header allow response to specify white-list, ask browser to only execute/render resource from white-list

PKI

Certificate: 1.A/B(PKA/B,PRKA/B, C(A/B), Certificate_CA) C(X)= PKX+PRKCA-signed[Hash(PKX+personInfoX)]

2.AB switch certificate, A verify C(B) **legitimate** should do: 2.1 dec C(B) with PKCA->get HASH1 2.2 signature algorithmn on (PKB+personInfoB) provided by C(B)->get HASH2 2.3 HASH1==HASH2→legitimate

3. AB switch certificate, A verify C(B) **belong to B** should do: enc HASH with PRKA to get signature, then dec sigature with PKB to get HASH'. HASH==HASH'→ belong B

Email Security

Email Security Threats related

Authenticity: result in unauthorized access to an email system
Integrity: result in unauthorized modification of email content
Confidentiality: result in unauthorize disclosure of sensitive information
Availability: prevent end users from able to send/receive email

S/MIME= Secure/Multipurpose Internet Mail Extension

Authentication=1. sender creates msg 2. use SHA-256 to generate 256-bit msg digest 3. encrypt msg digest with RSA using sender's PRK; append result as well as signer's identity to msg 4. receiver uses RSA with sender's PK to decrypt, recover, and verify msg digest

Confidentiality=1. sender create msg and random 128-bit number as a content-encryption key for this msg only 2. encrypt msg using content-encryption key 3. encrypt content-encryption key with RSA using receiver's PK and append it to msg 4. receiver use RSA with its PRK to decrypt and recover the content-encryption key 5. use content-encryption key to decrypt msg

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, 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/**OpenPGP public key server**; no vetting of OpenPGP keys, users decide whether to trust on their own

DANE allow X.509 certificate to be bound to DNS name using D NSSEC

TLSA Record=A new DNS record type defined by DANE
Used for secure method of authenticating SSL/TLS certificates
Specify constraints on which **CA** can vouch for a certificate, or w hich specific PKIX [Public Key Infrastructure (X.509)] end-entity c ertificate is valid
Specify service certificate / CA can be directly authenticate in DNS itself

DANE for SMTP
Targeted vulnerabilities: attackers can strip away TLS capability advertisement and downgrade the connection to not use TLS
TLS connections are often unauthenticated
A domain can use 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 name**
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 administrative domain fro m which email originates; at receiving end, MDA can access corre sponding public key via DNS and verify signature, thus authenticat ing that the message comes from claimed **administrative domain**
Difference from S/MIME and PGP: S/MIME and PGP use sender's private key to sign the content of the message; DKIM uses 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 use HMAC and X.509 digital certificate for authenticating traceback msgs. Router send ICMP traceback msgs to destination
Require all 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
1.Traceback from the router closest to victim
2.Determine upstream link that is used to carry out attack traffic
3.Recursively apply previous technique until attack source is reached

Input Debugging
1.Find attack signature(common feature contained in all attack packet)
2.Communicate attack signature to the upstream router, which then filters attack packets and determines the port of entry
3.Recursively apply the previous technique on the upstream routers until reaching the attack source
4.A considerable management overhead at the ISP level to communicate and coordinate the traceback

JY

Controlled Flooding
1.Need collaborative host and force them to flood links to upstream routers 2.Since buffer on victim is shared by all incoming links, flooding the link carrying out attack leads to drops of attack packets
3.Recursively apply previous technique on upstream router until reaching attack source 4.Require an accurate topology map. High overhead given multiple attacking sources (e.g., DDoS)

Logging-Based Traceback
1.Routers store packet logs 2.Victim queries closest routers about packet appearance of attack packets 3.router containing attack packet recursively query upstream routers until reaching attack source
Raw packets→high storage overhead on routers
Hash of invariant content per packet→high storage overhead given high traffic rate

Bloom Filter m-size bitmap, n members, k hash functions:
P(to get a false positive)=(1-(1-1/m)^(kn))^k

Network Protection

Firewall
Form a barrier through which 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(vice versa) must pass through 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 system; Collect information about attacker's activity; Encourage 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 data cleansing service to analyze traffic and filter malicious 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

JY

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 user possess for user authentication

Biometric: Authenticate a user based on 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 user's identity

ABAC=Attribute-Based Access Control Define authorizations that express conditions on properties of both the resource and the subject