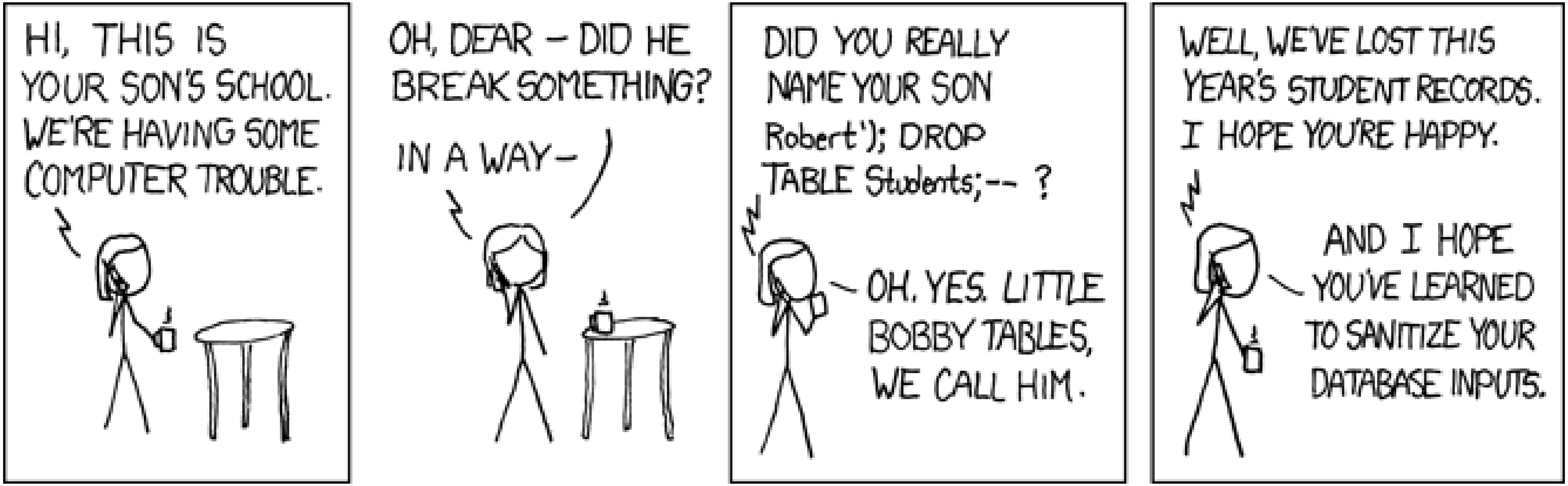
Web Application Security

Ivan Pepelnjak (@ioshints, ip@ipSpace.net)
NIL Data Communications

Security is like onion

Watching it makes you cry


HI, THIS IS YOUR SON'S SCHOOL. WE'RE HAVING SOME COMPUTER TROUBLE.
OH, DEAR – DID HE BREAK SOMETHING?
IN A WAY–
DID YOU REALLY NAME YOUR SON Robert'); DROP TABLE Students;-- ?
OH. YES. LITTLE BOBBY TABLES, WE CALL HIM.
WELL, WE'VE LOST THIS YEAR'S STUDENT RECORDS. I HOPE YOU'RE HAPPY.
AND I HOPE YOU'VE LEARNED TO SANITIZE YOUR DATABASE INPUTS.

Source: http://xkcd.com/327/

Original Sin

Security is an afterthought

- Fast and Fancy is more important than Secure
- Time-to-Market always wins
- Programmers are not security-aware (or have other targets)

Typical “solutions”

- Bolt-on security
- Belt-and-suspenders
- Trying to identify intrusions
  based on pattern matching

A Small Sample of Attack Types

Eavesdropping	Gain passive access to data in transit
Denial-of-Service	Regular users cannot access the service
Break-in	Attackers gain unauthorized access to systems
Data theft	Attackers gain unauthorized access to data
Data exfiltration	Unauthorized release of data
Defacement	Modification of public web sites
Malware propagation	Third-party web sites are used to propagate malware

http://en.wikipedia.org/wiki/Category:Computer_security_exploits

Typical Attacks

- Application
- Presentation
- Session
- Transport
- Network
- Data-link

SQL injection

Cross-site Scripting

Cross-site Request Forgery

Cookie hijacking

DNS hijacking

HTTP obfuscation

TCP SYN flood

TCP reset attack

Slow-motion attacks (slowloris)

Port scans

Packet flooding

ARP spoofing

Eavesdropping on shared media

Security Devices And Solutions

Packet filters

- Filters on 5-tuple (addresses, protocol, ports)

Stateful firewalls

- TCP session validation (prevents L2-L4 fuzzing)
- Dynamic session establishment

Deep packet inspection

- Inspect more than packet headers
- Required for badly-designed applications (FTP, SIP)
- Can be used for HTTP firewalling (not precise)

Web Application Firewalls

- Reassemble HTTP requests and responses
- Filters based on request/response content

Warning: Too much state will kill you

Firewall Versus IDS

Packet filters and firewalls

- Traffic not permitted is dropped
- Filtering rules are exact
- Mostly deterministic behavior
- DPI-based guessing introduces false positives

Intrusion Detection Systems

- All traffic is permitted
- IDS is passive, IPS/AV is active
- Pattern matching is used to identify evil payload or traffic pattern
- Alert is raised on pattern match
- Non-deterministic
- False positives
- Prone to obfuscation attacks

Sad fact: Most WAFs are configured in IDS/IPS mode

Security Then And Now

1990s

- Early commercial Internet deployments
- Misconfigured operating systems
- Highly vulnerable host stacks

Typical scenario:

- Attack: service or buffer overflow exploits
- Countermeasure: firewalls

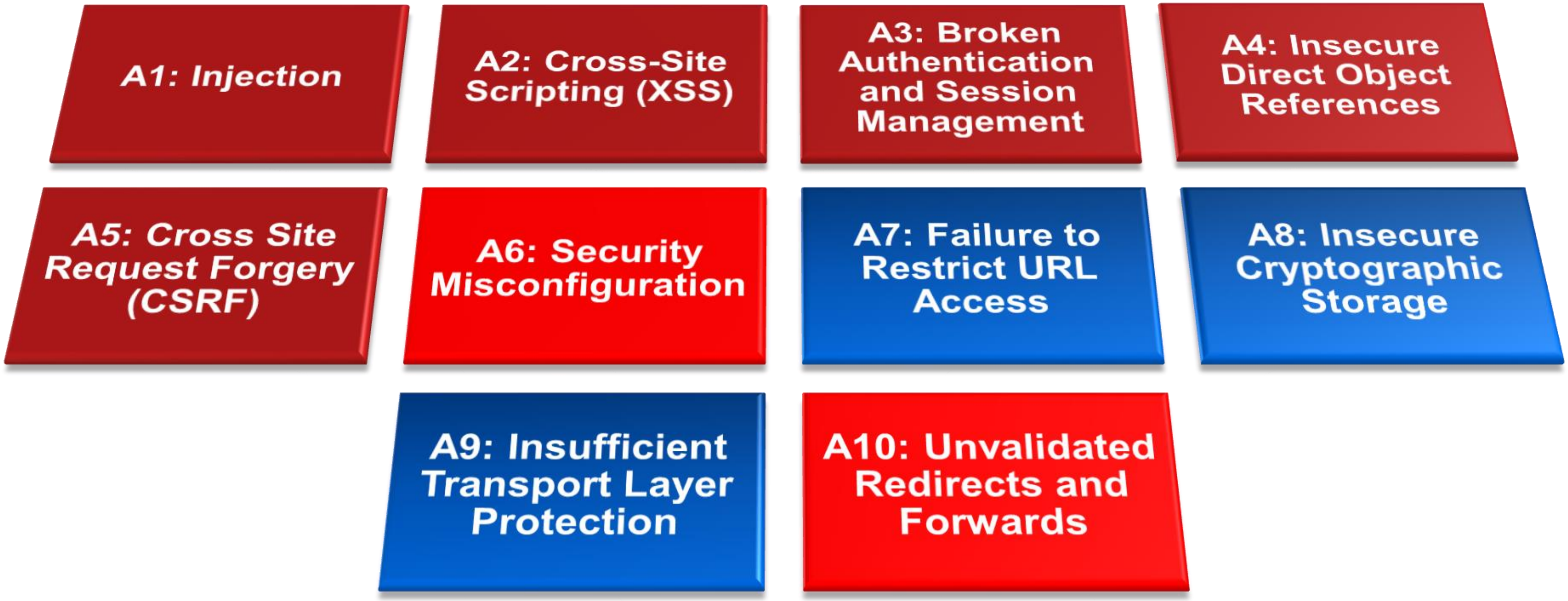
2010+

- Operating systems and TCP stacks fairly secure
- Web applications highly vulnerable (bad coding practices)

Typical scenario:

- Attack: **web application exploit**
- Countermeasure: WAF
- Firewalls and packet filters are basic hygiene

OWASP Top Ten



A1: Injection

A2: Cross-Site Scripting (XSS)

A3: Broken Authentication and Session Management

A4: Insecure Direct Object References

A5: Cross Site Request Forgery (CSRF)

A6: Security Misconfiguration

A7: Failure to Restrict URL Access

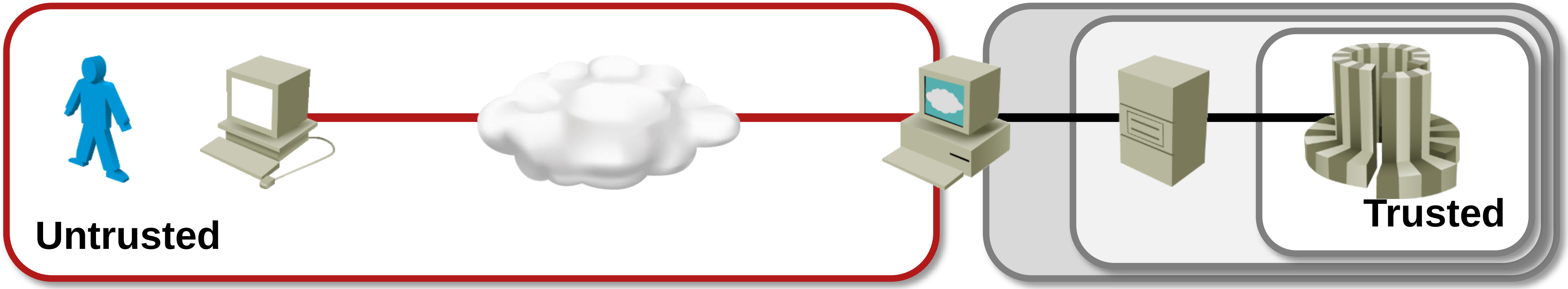
A8: Insecure Cryptographic Storage

A9: Insufficient Transport Layer Protection

A10: Unvalidated Redirects and Forwards

Source: http://www.owasp.org/index.php/Top_10

Web Application Trust Model



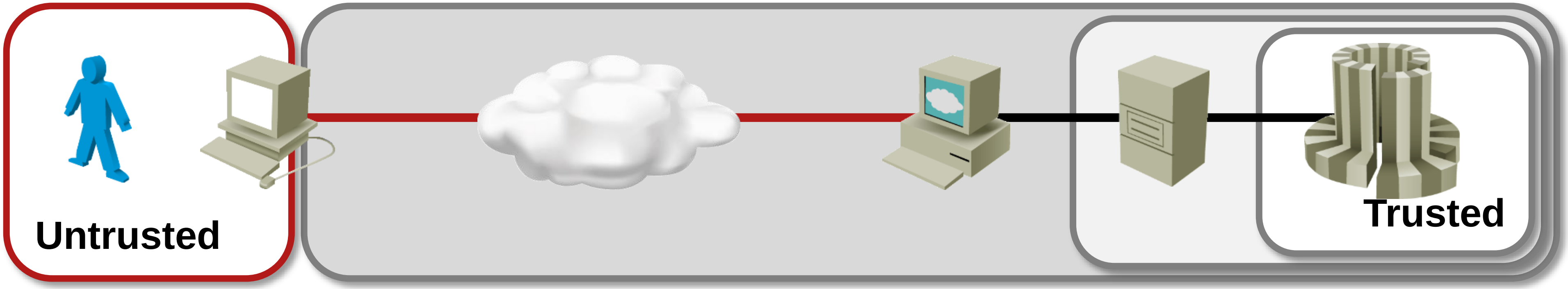
Web application perspective:

- Transport is untrusted
- Client is untrusted
- User is untrusted

User perspective:

- Transport is untrusted
- Web server might be fake

Increasing Trust: SSL (TLS)



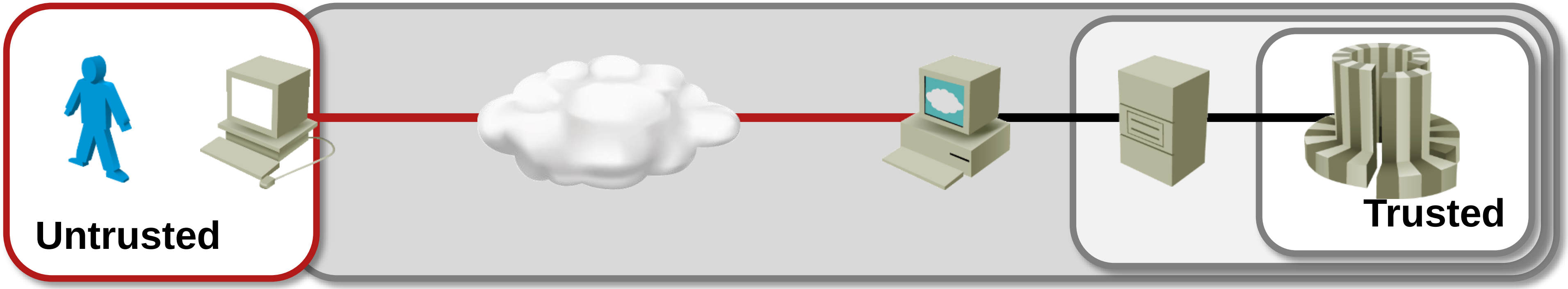
Web server perspective:

- Transport is trusted (encrypted)
- Client is untrusted
- User could be authenticated (client-side certificates)

User perspective:

- Transport is trusted
- Web server identity can be checked (modulo fake CA)

Never Trust the Web Client



The user can:

- Trigger fake requests
- Modify requests on-the-fly
- Modify/add/delete cookies
- View source code
- Modify DOM model

The browser/web app can:

- Trigger HTTP requests to any web site (CSRF)
- Mislead the user
- Hijack user clicks

The Fun Starts ...

Source: http://bit.ly/IOkTr9 (OWASP Top 10 Presentation)

Questions?