hkmconsulting, llc

Service Security

by Chris Riley

overview

- Web-based Services (SOAP / REST) challenge organizations in ways similar to web applications.
- Unlike web applications, service contracts provide simpler discoverability of potential attack vectors.
- Complex infrastructure and standards can reduce the application of deterrents.
- Traditional firewalls allow SOAP Traffic / REST Traffic to pass through over HTTP with no filtering.

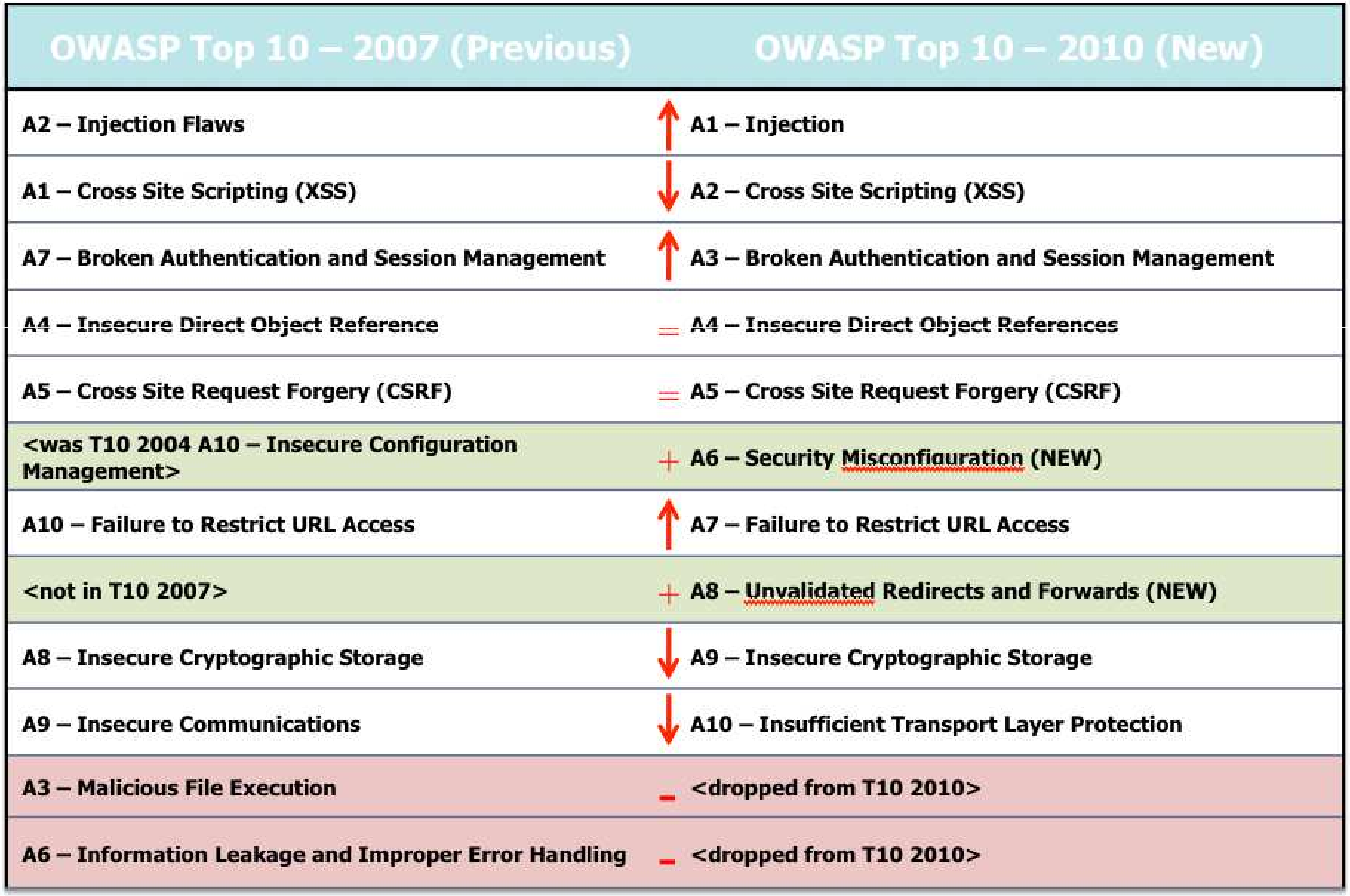
analysis vectors

1. Authentication – Who are you?
2. Authorization – What are you allowed to do?
3. Integrity – Is what your giving me forged or real?
4. Privacy/Confidentiality – Has the data been exposed?
5. Availability – Is the service available even under attack?
6. Logging – How do I find out what happened?

service security recommendations

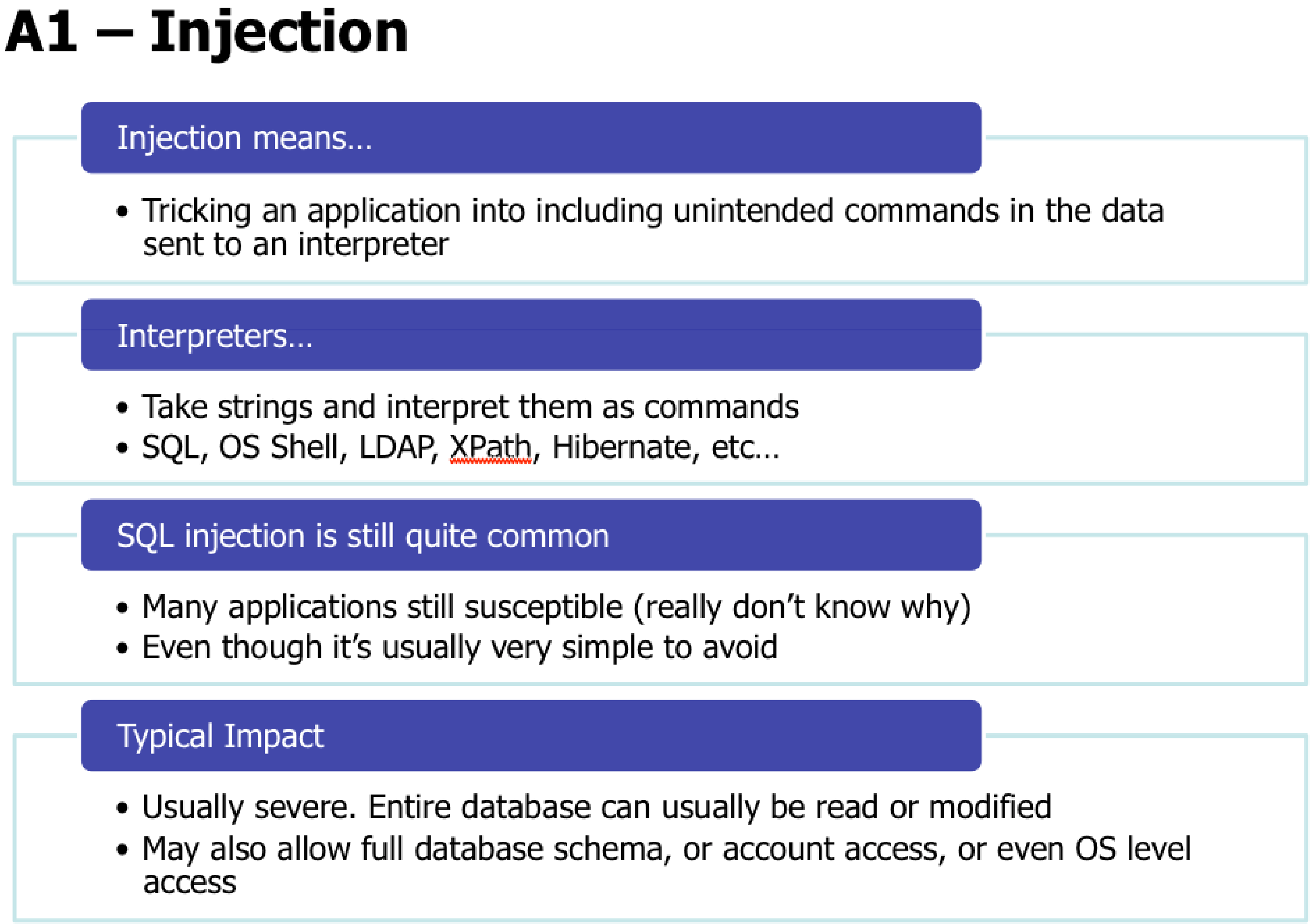
- OWASP
  - – *www.owasp.org/index.php/Web_Services#Securing_Web_Services*
- WS-I
  - – *www.ws-i.org/Profiles/BasicSecurity/SecurityChallenges-1.0.pdf*

OWASP 2007-2010 Top Ten



OWASP Top 10 – 2007 (Previous)		OWASP Top 10 – 2010 (New)
A2 – Injection Flaws	↑	A1 – Injection
A1 – Cross Site Scripting (XSS)	↓	A2 – Cross Site Scripting (XSS)
A7 – Broken Authentication and Session Management	↑	A3 – Broken Authentication and Session Management
A4 – Insecure Direct Object Reference	=	A4 – Insecure Direct Object References
A5 – Cross Site Request Forgery (CSRF)	=	A5 – Cross Site Request Forgery (CSRF)
<was T10 2004 A10 – Insecure Configuration Management>	+	A6 – Security Misconfiguration (NEW)
A10 – Failure to Restrict URL Access	↑	A7 – Failure to Restrict URL Access
<not in T10 2007>	+	A8 – Unvalidated Redirects and Forwards (NEW)
A8 – Insecure Cryptographic Storage	↓	A9 – Insecure Cryptographic Storage
A9 – Insecure Communications	↓	A10 – Insufficient Transport Layer Protection
A3 – Malicious File Execution	−	<dropped from T10 2010>
A6 – Information Leakage and Improper Error Handling	−	<dropped from T10 2010>

OWASP Attack Vector Example



A1 – Injection

Injection means…

- Tricking an application into including unintended commands in the data sent to an interpreter

Interpreters…

- Take strings and interpret them as commands
- SQL, OS Shell, LDAP, XPath, Hibernate, etc…

SQL injection is still quite common

- Many applications still susceptible (really don’t know why)
- Even though it’s usually very simple to avoid

Typical Impact

- Usually severe. Entire database can usually be read or modified
- May also allow full database schema, or account access, or even OS level access

WS-I Security Challenges 2005-07

- Threats
  - *T-01: Message Alteration*
  - *T-02: Confidentiality*
  - *T-03: Falsified Messages*
  - *T-04: Man in the Middle*
  - *T-05: Principal Spoofing*
  - *T-06: Forged Claims*
  - *T-07: Replay of Message Parts*
  - *T-08: Replay*
  - *T-09: Denial of Service*
  - *T-(OOS)-XX: Out of Scope Threats (Another 14)*

WS-I.org Challenges Example

- Threat Associations are mapped to potential solutions / analysis vectors:

3.3 C-04: Data Confidentiality Definition: Data confidentiality: The property that information is not made available or disclosed to unauthorized individuals, entities, or processes [i.e. to any unauthorized system entity].

Explanation: The property that eavesdroppers or other unauthorized parties cannot view confidential message content. Typically this is achieved with encryption. Note that confidentiality is a distinct concept from privacy, so in the definition "disclosure" refers to the ability to view or eavesdrop the information when transferred or processed. Confidentiality techniques may be used as one aspect of maintaining privacy, however.

Threat Associations: T-02, T(OOS)-10, T(OOS)-14. Disclosure related attacks as well as attacks that reduce the confidentiality strength (e.g. man-in-the-middle SSL/TLS cipher suite attacks) are relevant.

Available at: http://www.ws-i.org/profiles/basicsecurity/securitychallenges-1.0.pdf

service security options

- Transport Security
  - Focuses on the IP Layer between a consumer and provider.
  - REST employs this strategy for security
  - Administrators are well versed and strategies are common.
- Message Security
  - Focuses on securing the message instead of the transport.
  - SOAP-based Web Services employ this strategy
  - Security is maintained across intermediaries/transports.
  - Not as well known and more difficult to administer.

security standards

- W3C (XML Encryption, XML-DigSig)
- OASIS (WS-Security, SAML, XACML)
- IETF – SSL/TLS

	Transport-level	**Message-level**
Authentication	Basic/Digest Client Authentication	UsernameToken XML Signature
Authorization	Custom	SAML/XACML
Confidentiality / Privacy	SSL / TLS	XML Encryption
Integrity/Non-Repudiation	SSL / TLS	XML Signature
Single Sign-On	Custom	SAML

hkmconsulting,llc

Hacking Scenarios

blackbox assessment

- Assessment from the Hacker's perspective
- Probe for service endpoints to gain access to WSDL/XSD.
  - UDDI Query / Public Search (inurl:wsdl site:ebay.com)
  - Crawling (wget -l 50 -r http://server)
  - Directory Traversal Attacks (identify endpoint and then use parent directory to see if other resources are exposed)

blackbox assessment

- Identify server platform
  - HTTP HEAD request along with URL exposes deployment platform details (.Net, Axis etc.)
- Scan WSDL / XSD to identify operations, messages, elements and data constraints.
  - Comments/annotations may hint at platform and known issues with service quality

whitebox assessment

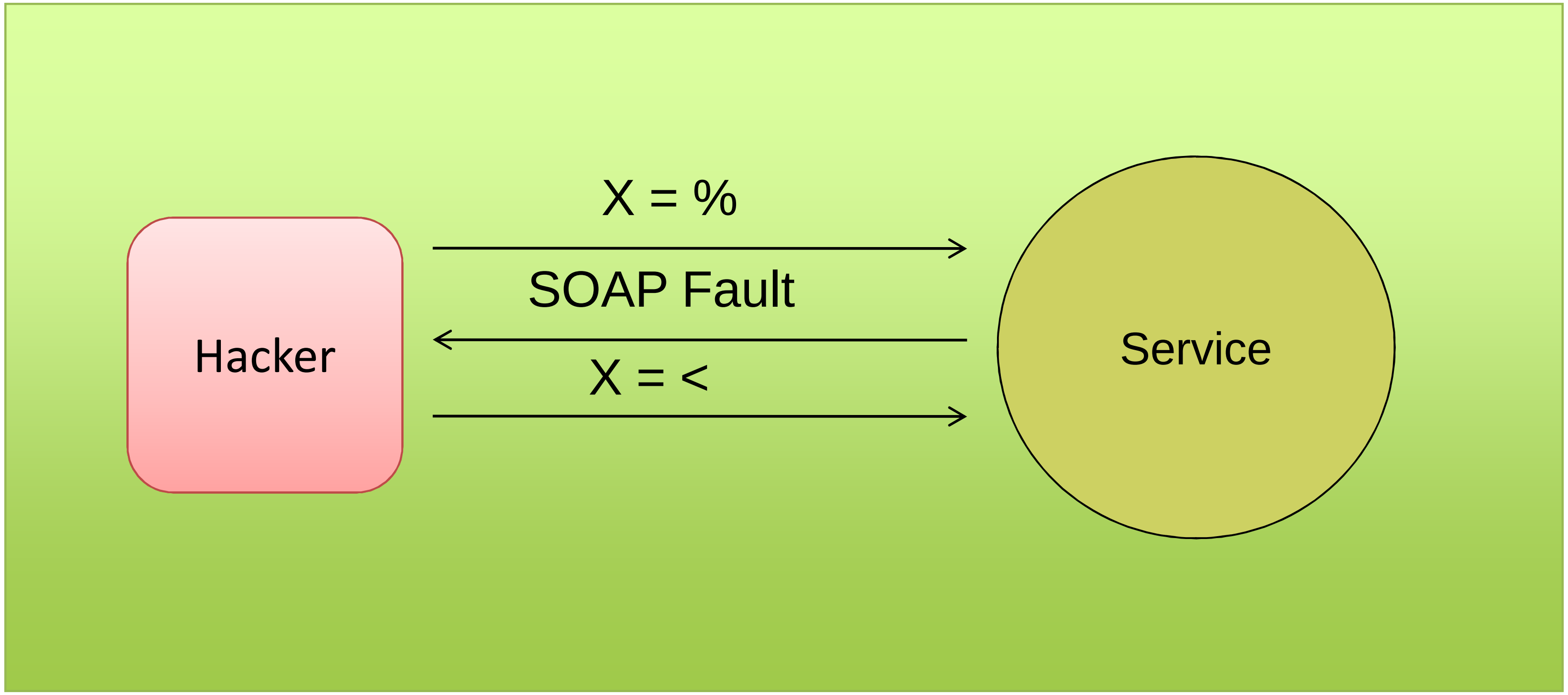
- Examine the service with full knowledge of the environment and service.
- Assess the service details in-transit, the server and the service core logic.
- Verification that delivered service is following excepted security design standards, design specifications via a methodical testing process.

attack vectors

- Parameter Tampering
- Injection (SQL/XPath) – **A1**
- Denial of Service / Distributed Denial of Service – **T-09, T(OOS)-11, T(OOS)-12**
- Replay – **T-07, T-08**
- WSDL Spoofing - **T-04**
- XML Poisoning – **T-01, T-03**
- Improper Security Configuration – **A6, T(OOS)-14**

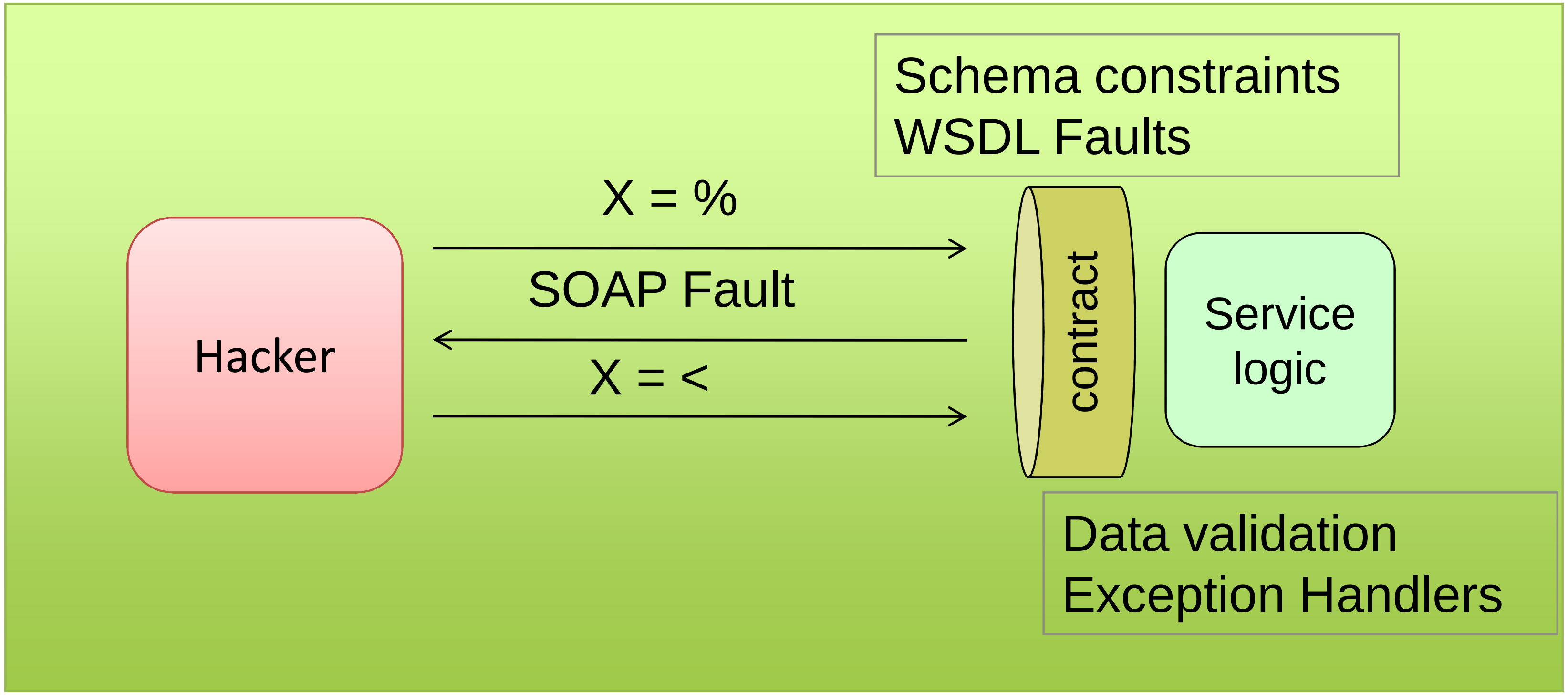
parameter tampering

- Goal: Probe Web Service with variations of parameters to gain further details via SOAP/Server Faults.



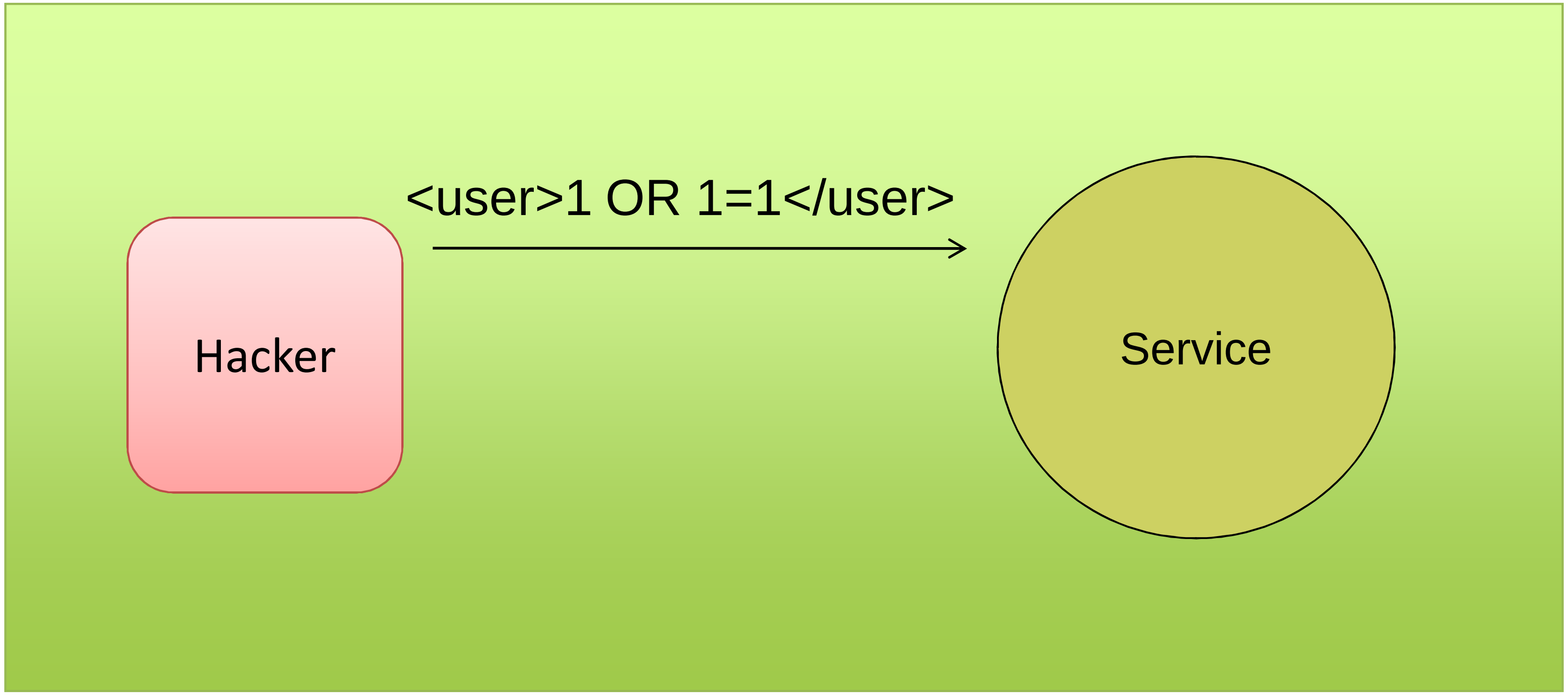
parameter tampering

- Solution: Proper application of exception handling, finer constraint granularity and data validation to increase Service Abstraction. Also referred to as Content Filtering.



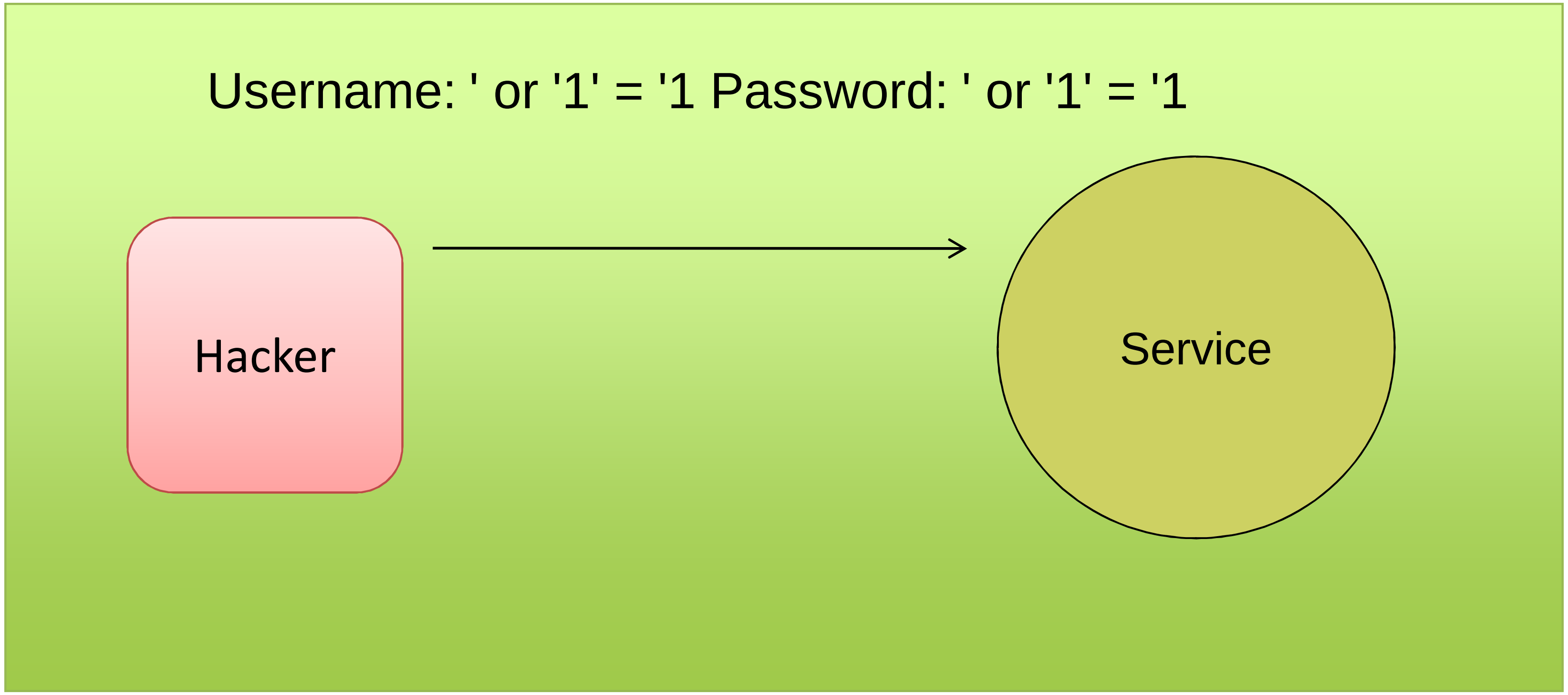
sql injection

- Goal: Inserting malicious SQL queries into user input to access/manipulate data in the database.



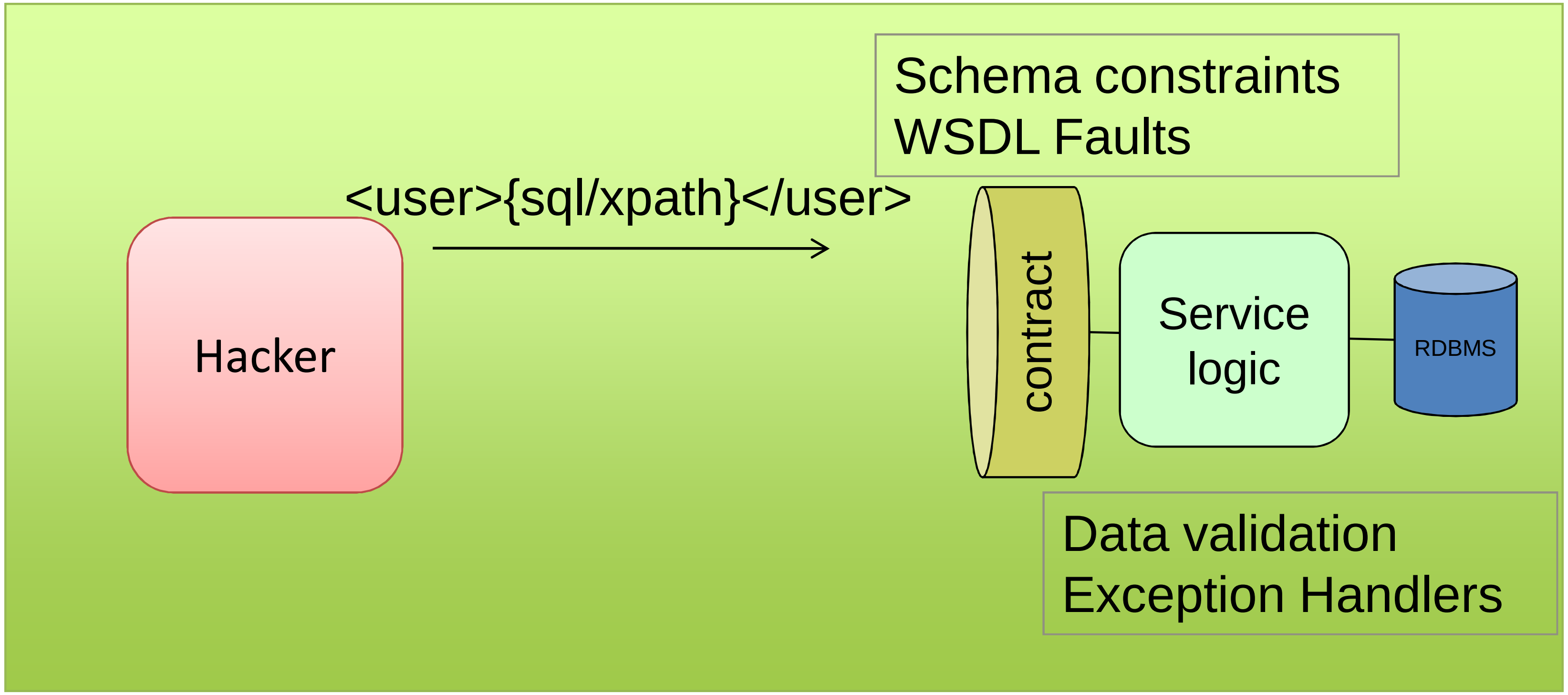
xpath injection

- Goal: Inject data into queries to allow for user control. For example altering XPath to always evaluate to true when evaluating credentials.



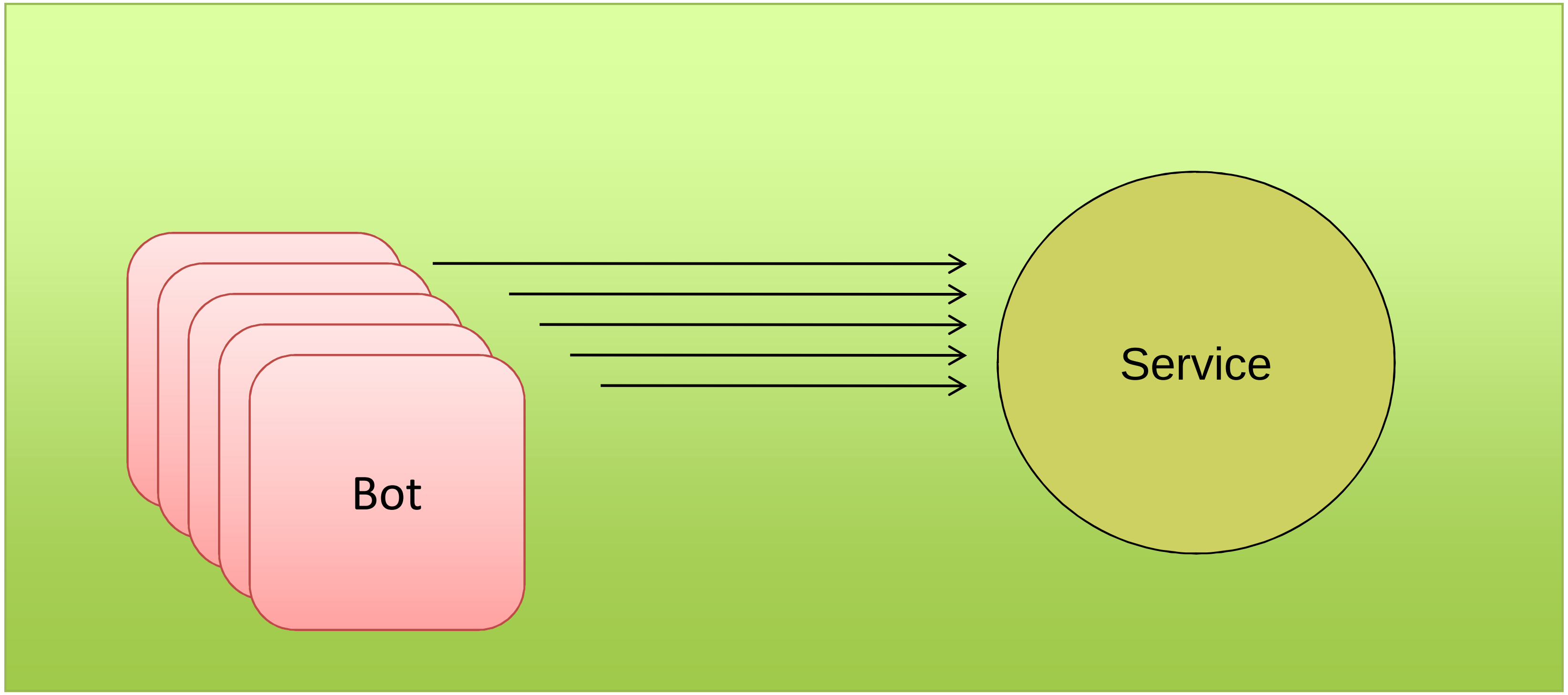
injection

- Solution: Proper application of exception handling, finer constraint granularity and data validation to reduce malicious queries, informative responses and errors.



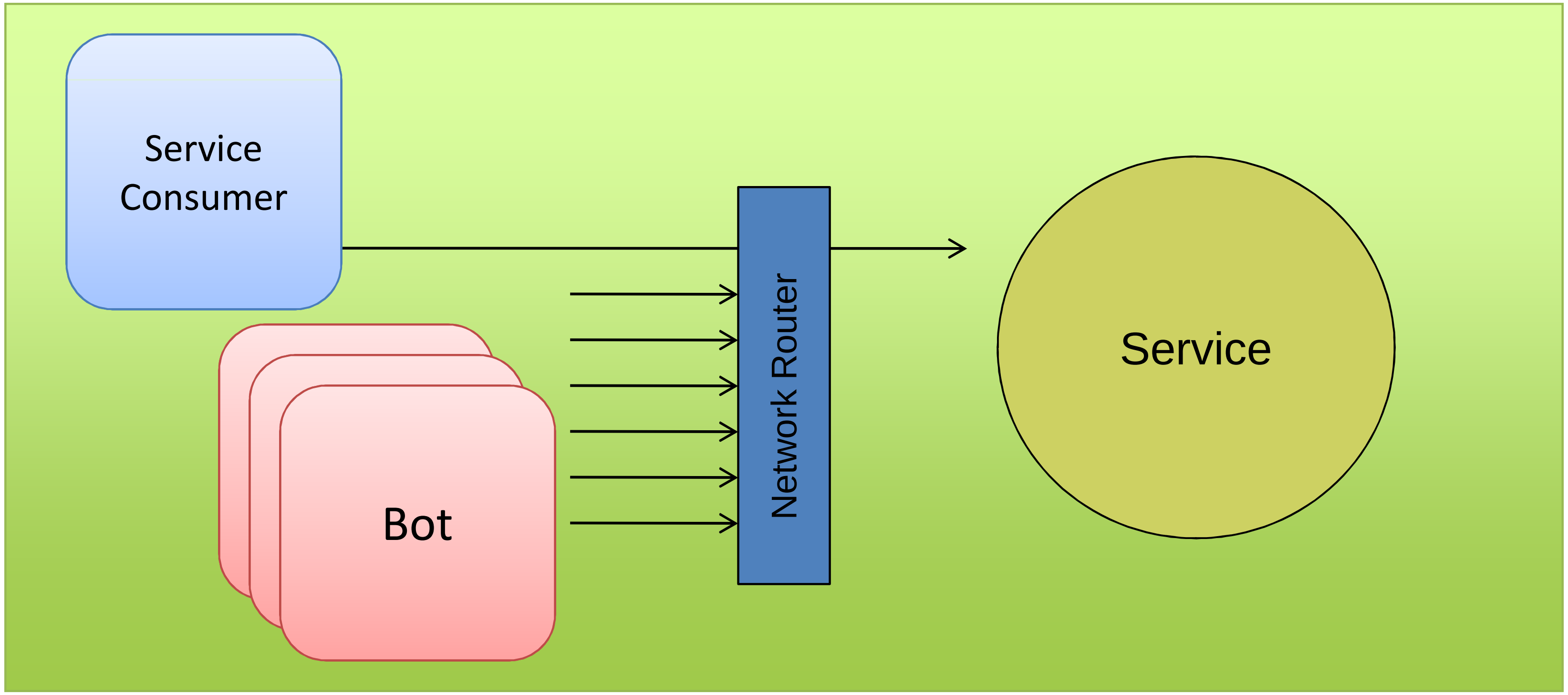
Denial of Service – DoS / DDoS

- Goal: Coordinated attack of an endpoint by flooding with numerous requests exceeding server resources.



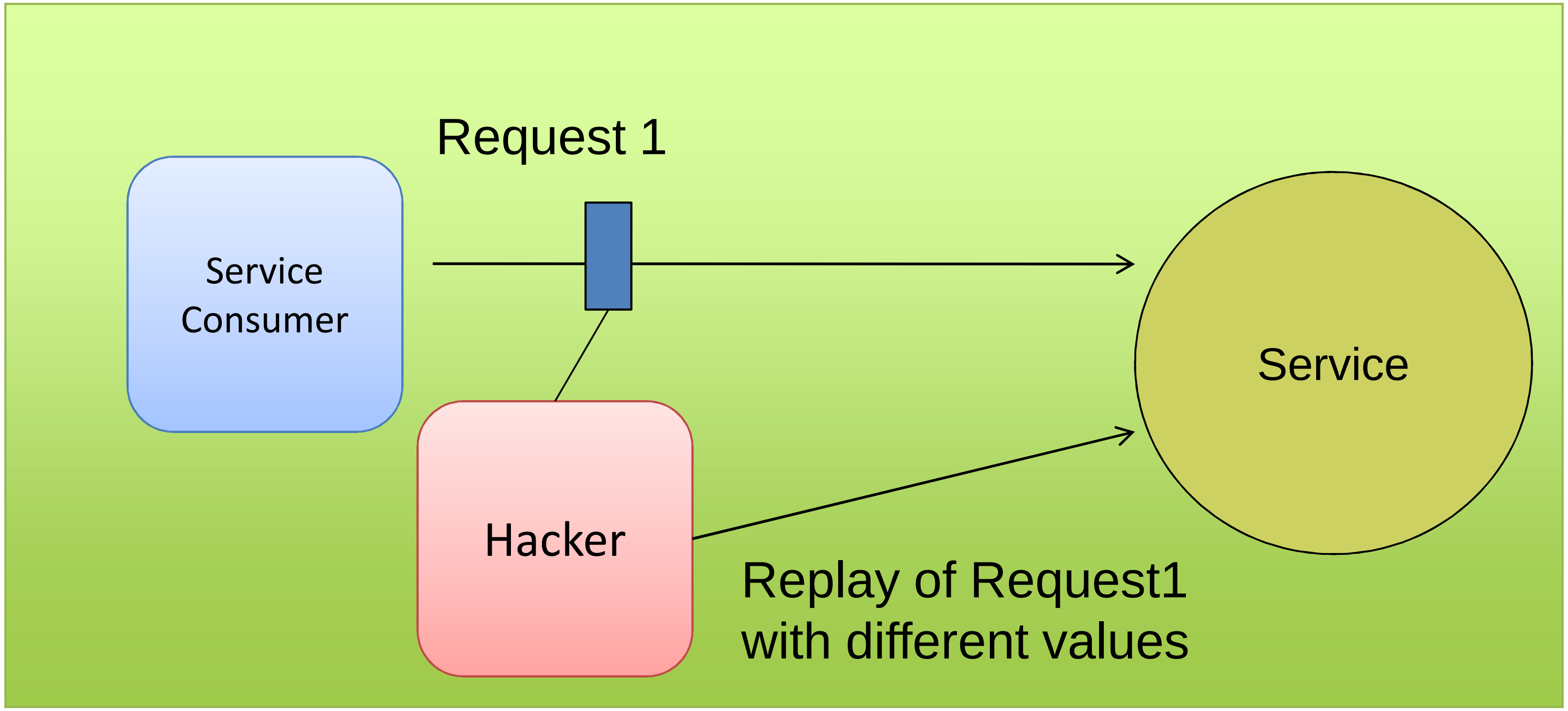
Denial of Service – DoS / DDoS

- Solution: Use of Rate Limiting within Routers, application allowable consumer IP addresses, network redundancy, geographically diverse networks, patching of systems.



replay

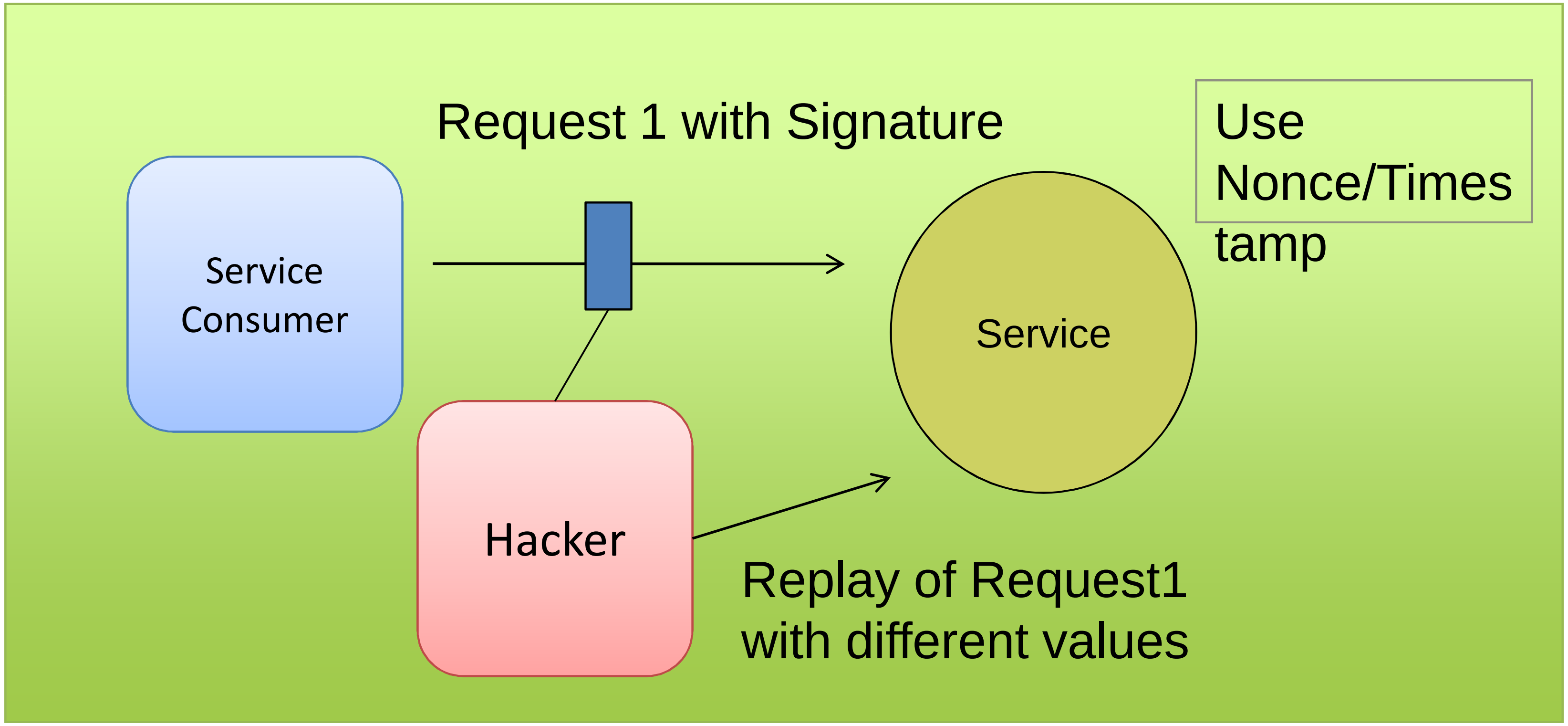
- Goal: A request is recorded/intercepted and reused to affect a different result. Request will replay the authentication details.



hkmconsulting, llc

replay

- Solution: A Signature, Nonce (unique generated value) and Timestamp can be utilized to and confirm uniqueness.

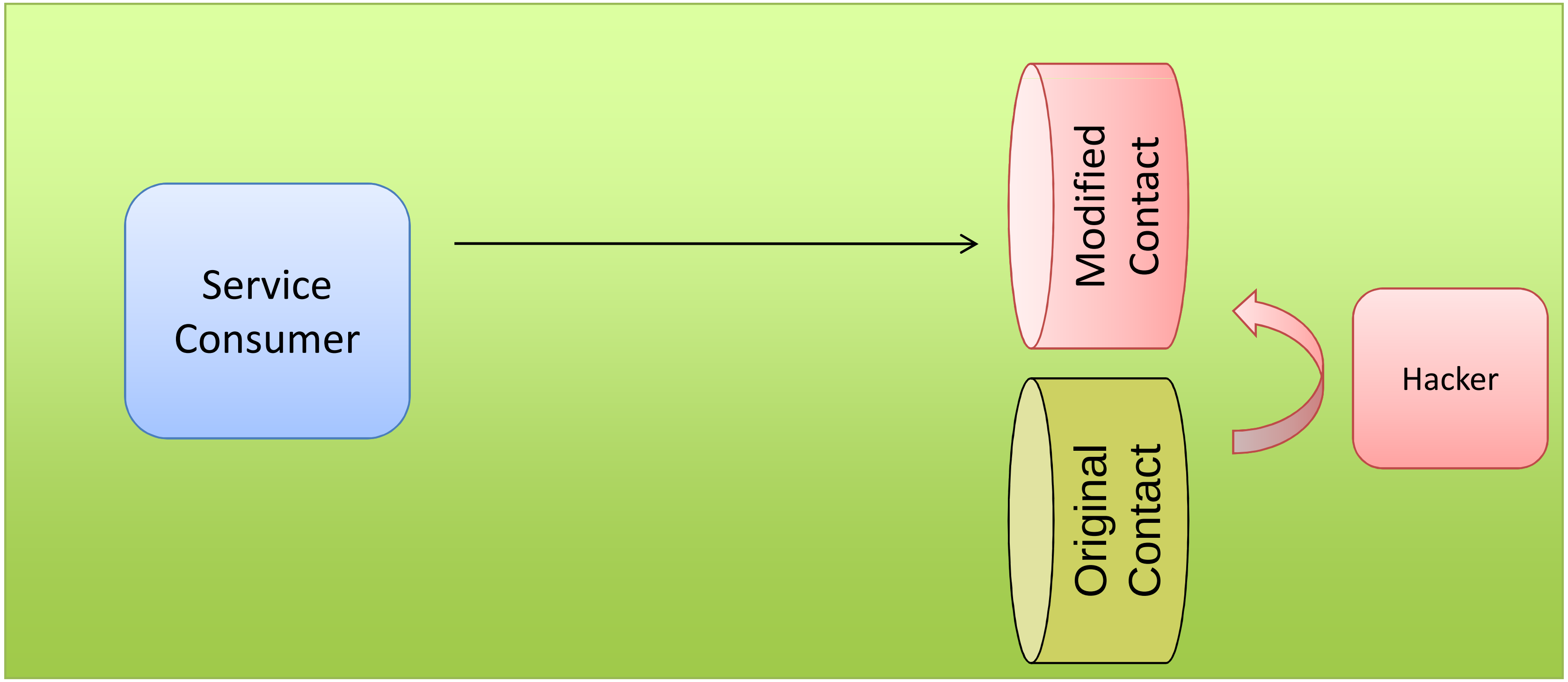


xml poisoning

- Goal: XML is constructed to cause spoiling of data or excessive parsing of content.
- SAX and DOM Parsers provide two alternative mechanisms for processing XML.
  - XML external entity references are used to open up files/connections to other resources for leveraging an attack.
  - Circular references and Large XML payloads can cause excessive processing.
- Solution: Use of request filtering/interceptors to isolate before parsing such as XML Gateway/Firewalls.

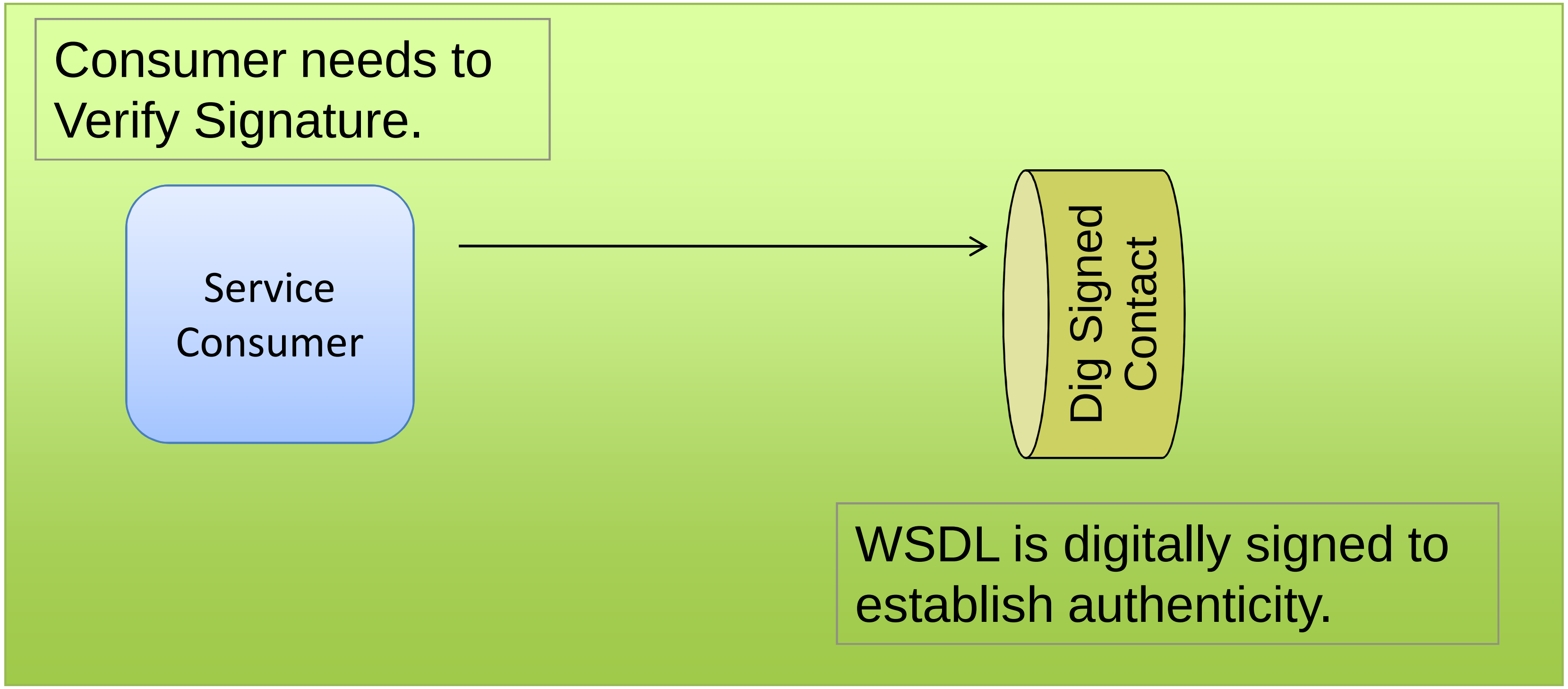
wsdl spoofing

- Goal: Service contracts could be altered or replaced with a fake WSDL definition to spoof a site unbeknownst to the service consumer.



wsdl spoofing

- Solution: WSDL is digitally signed to confirm authenticity of definition. Service consumer needs to examine signature prior to utilization.



Consumer needs to Verify Signature.

WSDL is digitally signed to establish authenticity.

security configuration

- Goal: Identification of target environment details to exploit older versions of service platforms and known vulnerabilities.
- Solution: Awareness of platform security defects and consistent application of patches/upgrades to reduce the likelihood of exploitation.

- Cloud Computing introduces some new threat vectors:
  - – Poisoned Amazon Machine Instance (AMI)
    - Beware of community images, make your own
  - – Amazon Management Console Attacks
    - Vulnerable due to Amazon.com domain
    - Credentials are Amazon.com versus AWS
    - Console and Web Services allow for deletion/manipulation of the deployed infrastructure.

Summary

- In-transit
  - IP Firewall restrictions / limits / redundancy
  - Security (Encryption, Authentication, Authorization…)
- Service Container (Discoverability Principle)
  - Disable WSDL access, Sign Service Contract
  - Disable responses/debug output for security violations
  - Enable Security Auditing/Logging
- Service (Abstraction Principle)
  - Constraints / Data Validation
  - Exception Handling
- Platform patching / configuration

references

- Hacking, The Next Generation (2009), O’Reilly
- SOA Security, (2008), Manning
- Hacking Web Services, (2007), Delmar Cengage Learning
- Web Services Security, (2003), McGraw-Hill
- OWASP Top Ten
- WS-I Security Challenges