Sicurezza Applicativa

Prof. Francesco Bergadano

Dipartimento di Informatica
Università degli Studi di Torino

Principali vulnerabilità applicative

1. **La OWASP Foundation**
2. **OWASP “top-ten application security risks”**

OWASP Foundation

1. **“Open Web Application Security Project”, Fondazione no-profit**
2. **Community, basata su lavoro volontario, indipendente**
3. **Risultati, pubblicazioni, metodologie open: top ten vulnerabilities, testing methodology, software**

OWASP Top Ten

1. **Injection**
2. **XSS (cross site scripting)**
3. **Broken Authentication & session management**
4. **Insecure direct object references**
5. **CSRF (cross site request forgery)**
6. **Security misconfiguration**
7. **Insecure cryptographic storage**
8. **Failure to restrict URL access**
9. **Insufficient transport layer protection**
10. **Unvalidated redirects and forwards**

OWASP Top Ten

1. **Injection**
2. **XSS (cross site scripting)**
3. **Broken Authentication & session management**
4. **Insecure direct object references**
5. **CSRF (cross site request forgery)**
6. **Security misconfiguration**
7. **Insecure cryptographic storage**
8. **Failure to restrict URL access**
9. **Insufficient transport layer protection**
10. **Unvalidated redirects and forwards**

Injection

Risulta possibile inviare all'applicazione codice arbitrario, che poi viene eseguito lato server:

Esempio 1: SQL Injection

Esempio 2: Buffer Overflow

OWASP Top Ten

1. **Injection**
2. **XSS (cross site scripting)**
3. **Broken Authentication & session management**
4. **Insecure direct object references**
5. **CSRF (cross site request forgery)**
6. **Security misconfiguration**
7. **Insecure cryptographic storage**
8. **Failure to restrict URL access**
9. **Insufficient transport layer protection**
10. **Unvalidated redirects and forwards**

XSS – Cross site scripting

Uno script può essere eseguito sul Browser della vittima, facendo in modo che i parametri del Browser relativi al sito target (in particolare I cookies) vengano intercettati.

OWASP Top Ten

1. **Injection**
2. **XSS (cross site scripting)**
3. **Broken Authentication & session management**
4. **Insecure direct object references**
5. **CSRF (cross site request forgery)**
6. **Security misconfiguration**
7. **Insecure cryptographic storage**
8. **Failure to restrict URL access**
9. **Insufficient transport layer protection**
10. **Unvalidated redirects and forwards**

Broken Authentication & session management

Broken authentication: l'autenticazione utente viene resa vana, ovvero un utente non autorizzato riesce ad autenticarsi.

Session management: un utente non autorizzato riesce ad inserirsi all'interno di una sessione di un utente autorizzato.

Broken Authentication

Esempio:

- **l'utente legittimo si autentica su un sito Web in modo sicuro, il sito mantiene la sessione nella URL, e richiede vari input (es. carta di credito, codici di promozione).**
- **Lo stesso utente invia la URL ad altri, per esempio per informare su un'offerta vantaggiosa.**
- **Questi altri usano la URL per proseguire la sessione e acquistare altri beni con i dati del primo utente.**

Session management

Esempio:

- **Una sessione viene gestita via cookies senza scadenza, e rimane attiva senza time-out.**
- **Non vi è un bottone di logout, o l'utente non lo usa.**
- **Immaginando che l'utente abbia usato una postazione pubblica, un altro utente può usare dopo la stessa postazione, proseguendo la sessione del primo utente.**

OWASP Top Ten

1. **Injection**
2. **XSS (cross site scripting)**
3. **Broken Authentication & session management**
4. **Insecure direct object references**
5. **CSRF (cross site request forgery)**
6. **Security misconfiguration**
7. **Insecure cryptographic storage**
8. **Failure to restrict URL access**
9. **Insufficient transport layer protection**
10. **Unvalidated redirects and forwards**

Insecure direct object references

Un oggetto è raggiungibile all'interno di una sessione, dopo l'autenticazione, es. http://www.sito.it/path?account=xy_113

Un altro oggetto risulta raggiungibile anche dopo, anche da un altro utente, usando la stessa URL con parametri modificati facilmente indovinabili, es. http://www.sito.it/path?account=xy_117

OWASP Top Ten

1. **Injection**
2. **XSS (cross site scripting)**
3. **Broken Authentication & session management**
4. **Insecure direct object references**
5. **CSRF (cross site request forgery)**
6. **Security misconfiguration**
7. **Insecure cryptographic storage**
8. **Failure to restrict URL access**
9. **Insufficient transport layer protection**
10. **Unvalidated redirects and forwards**

Cross site request forgery

Un sito permette operazioni critiche agli utenti autorizzati previa autenticazione, durante la stessa sessione, es. www.sito.it/bonifico?euro=200&conto=134

**Lo stesso utente autorizzato apre una pagina fraudolenta in un'altra scheda, che lo porta ad richiedere una diversa operazione, all'interno della sessione autenticata: **

Cross site request forgery

Esempio:

1. **Fare directory con basic authentication con Apache**
2. **Autenticarsi**
3. **Aprire altra scheda, risulteremo già autenticati**

OWASP Top Ten

1. **Injection**
2. **XSS (cross site scripting)**
3. **Broken Authentication & session management**
4. **Insecure direct object references**
5. **CSRF (cross site request forgery)**
6. **Security misconfiguration**
7. **Insecure cryptographic storage**
8. **Failure to restrict URL access**
9. **Insufficient transport layer protection**
10. **Unvalidated redirects and forwards**

Security misconfiguration

Ad esempio:

Il framework di sviluppo non è aggiornato

Installo un pacchetto lato server ma non cambio gli utenti di default (es. user admin, password admin)

Directory listing in Apache

OWASP Top Ten

1. **Injection**
2. **XSS (cross site scripting)**
3. **Broken Authentication & session management**
4. **Insecure direct object references**
5. **CSRF (cross site request forgery)**
6. **Security misconfiguration**
7. **Insecure cryptographic storage**
8. **Failure to restrict URL access**
9. **Insufficient transport layer protection**
10. **Unvalidated redirects and forwards**

Insecure cryptographic storage

Ad esempio:

Dati riservati sono memorizzati in chiaro, oppure la chiave è memorizzata in chiaro, sul server o sui backup

Password e chiavi crittografiche deboli rispetto a guessing e cracking

Password “unsalted”

OWASP Top Ten

1. **Injection**
2. **XSS (cross site scripting)**
3. **Broken Authentication & session management**
4. **Insecure direct object references**
5. **CSRF (cross site request forgery)**
6. **Security misconfiguration**
7. **Insecure cryptographic storage**
8. **Failure to restrict URL access**
9. **Insufficient transport layer protection**
10. **Unvalidated redirects and forwards**

Failure to restrict URL access

Un utente, previa autenticazione, può raggiungere una certa URL, es.

www.sito.it/dati_pubblici.html

Lo stesso utente, o un altro, riesce ad indovinare una URL simile, che però non risulta protetta, es.

www.sito.it/dati_privati.html

OWASP Top Ten

1. **Injection**
2. **XSS (cross site scripting)**
3. **Broken Authentication & session management**
4. **Insecure direct object references**
5. **CSRF (cross site request forgery)**
6. **Security misconfiguration**
7. **Insecure cryptographic storage**
8. **Failure to restrict URL access**
9. **Insufficient transport layer protection**
10. **Unvalidated redirects and forwards**

Insufficient transport layer protection

SSL solo su alcune delle pagine o componenti riservate.

Certificati scaduti o con CA non riconosciute.

In generale, possibilità di intercettare il traffico (es. Arp / DNS spoofing)

OWASP Top Ten

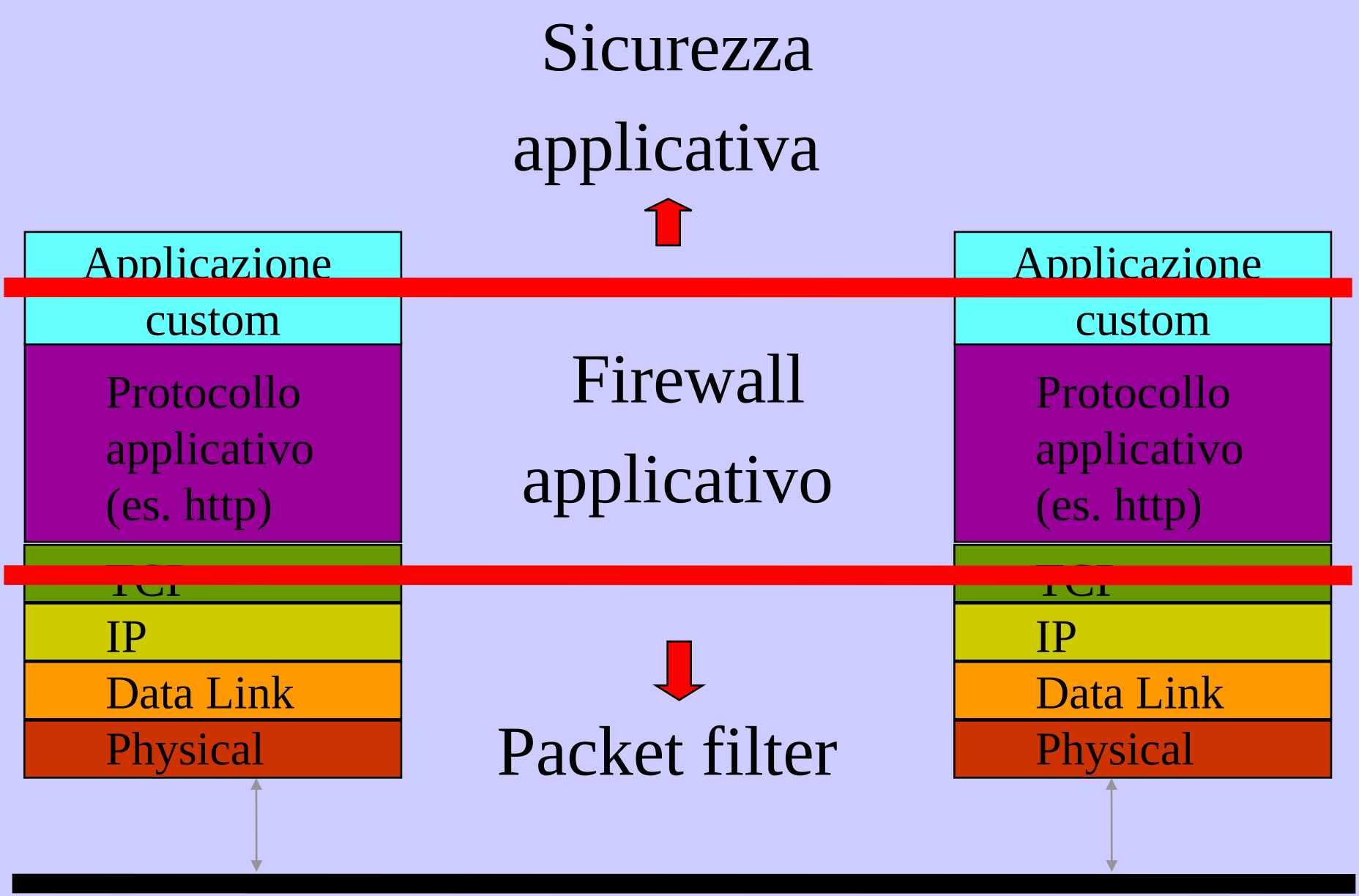
1. **Injection**
2. **XSS (cross site scripting)**
3. **Broken Authentication & session management**
4. **Insecure direct object references**
5. **CSRF (cross site request forgery)**
6. **Security misconfiguration**
7. **Insecure cryptographic storage**
8. **Failure to restrict URL access**
9. **Insufficient transport layer protection**
10. **Unvalidated redirects and forwards**

Unvalidated redirects and forwards

Es.:

L’utente prevede delle pagine che ridirezionano l’utente verso una URL ottenuta a partire da un parametro – l’utente può cliccare su un link che lo porta al sito, ma poi lo ridireziona su un sito pericoloso.

Stessa cosa per i forward interni


Sicurezza
applicativa
Applicazione
custom
Protocollo
applicativo
(es. http)
IP
Data Link
Physical
Firewall
applicativo
Applicazione
custom
Protocollo
applicativo
(es. http)
IP
Data Link
Physical
Packet filter

OWASP top-ten:

un buon punto di partenza, da abbinarsi a best practices, test in collaudo e in produzione, uso di framework

Altre vulnerabilità

DOS (OWASP top ten – entry A9 – nel 2004), es.:

- **syn flooding**
- **“smurf”**

Insufficient Anti-automation, es.:

- **biometrics**
- **captcha**

Phishing

Metodi di autenticazione per online banking

es.:

- **one time passwords**
- **password dispositive**
- **SSL client authentication**
- **autenticazione con firma elettronica**

Metodologia OWASP:

come valutare le vulnerabilità in termini di rischio e business impact

Metodologia OWASP:

threat agents – attack vectors – security weaknesses – tech impacts – business impact

OWASP Top Ten

1. **Injection**
2. **XSS (cross site scripting)**
3. **Broken Authentication & session management**
4. **Insecure direct object references**
5. **CSRF (cross site request forgery)**
6. **Security misconfiguration**
7. **Insecure cryptographic storage**
8. **Failure to restrict URL access**
9. **Insufficient transport layer protection**
10. **Unvalidated redirects and forwards**

OWASP Top Ten Risk Analysis

	Threat agent	Attack vectors	Weakness Prevalence	Weakness Detectability	Tech impact	Business impact
1		easy	common	average	severe	
2		average	very widespread	easy	moderate	
3		average	common	average	severe	
4		easy	common	easy	moderate	
5		average	widespread	easy	moderate	
6		easy	common	easy	moderate	
7		difficult	uncommon	difficult	severe	
8		easy	uncommon	average	moderate	
9		difficult	common	easy	moderate	
10		average	uncommon	easy	moderate	

Metodologia OWASP:

l'analisi del rischio permette di:
- *stimare l'extra effort di sviluppo*
- *decidere se effettuare test*
- *decidere se dismettere/aggiornare*