

Linee guida per lo sviluppo sicuro

SOMMARIO

1	INTRODUZIONE	7
1.1	SCOPO	7
1.2	STRUTTURA DEL DOCUMENTO	7
2	RIFERIMENTI	8
2.1	DOCUMENTI DI RIFERIMENTO	8
3	DEFINIZIONI E ACRONIMI	9
3.1	DEFINIZIONI	9
3.2	ACRONIMI	9
4	SVILUPPARE APPLICAZIONI SICURE	12
5	PROGETTAZIONE E SVILUPPO DELL'APPLICAZIONE: DIRETTIVE STANDARD	13
5.1	PROGETTAZIONE DELL'APPLICAZIONE	13
5.2	SVILUPPO DELL'APPLICAZIONE – CRITERI GENERALI	13
5.2.1	<i>Performance</i>	13
5.2.2	<i>Password nel codice sorgente</i>	14
5.2.3	<i>Privilegi esecutivi minimi</i>	14
5.2.4	<i>Metodi TRACE e TRACK</i>	14
5.2.5	<i>Assenza di codice malevolo</i>	14
5.2.6	<i>Fattore integrità</i>	14
5.2.7	<i>Input character validation</i>	14
5.2.8	<i>Gestione dell'output</i>	15
5.3	FORMATTAZIONE DEL CODICE	15
5.3.1	<i>Stile e sintassi</i>	15
5.3.2	<i>Algoritmi</i>	16
5.3.3	<i>Utilizzo funzioni di gestione delle stringhe</i>	16
5.3.4	<i>Specifiche del formato delle stringhe</i>	16
5.3.5	<i>Casting e variabili numeriche</i>	16
5.4	TRACCIAMENTO E RACCOMANDAZIONI DI "ALARM DETECTION"	16
5.4.1	<i>Tracciamento eventi</i>	16
5.4.2	<i>Tracciamento eventi di "Alarm Detection"</i>	17
5.4.3	<i>Scopo e campo di applicazione per eventi di "Alarm Detection"</i>	17
5.4.4	<i>Raccomandazioni generali per eventi di "Alarm Detection"</i>	17
5.5	COMPILAZIONE DELL'APPLICAZIONE	18
5.5.1	<i>Stack Canary</i>	18
5.5.2	<i>Correttezza del sorgente</i>	18
5.6	AMBIENTE OPERATIVO DELL'APPLICAZIONE	18
5.6.1	<i>Separazione degli ambienti</i>	18
5.6.2	<i>Test dell'Applicazione</i>	18
5.6.3	<i>Strumenti</i>	19
5.6.4	<i>Profili utente</i>	19
5.6.5	<i>Trattamento dei dati</i>	19
5.6.6	<i>Protezione dei sorgenti e delle librerie</i>	19
5.7	AUTENTICAZIONE, AUTORIZZAZIONE E GESTIONE DEGLI ACCESSI	19
5.7.1	<i>Policy standard "Everything is generally forbidden unless expressly permitted"</i>	19
5.7.2	<i>Assegnazione dei privilegi utente</i>	19
5.7.3	<i>Procedura di accesso dell'applicazione</i>	19
5.7.4	<i>Account standard</i>	20
5.7.5	<i>Autorizzazione</i>	20
5.7.6	<i>Generazione dei token</i>	20
5.7.7	<i>Generazione dei cookie</i>	20
5.7.8	<i>Contenuto del cookie</i>	20
5.7.9	<i>Scadenza del cookie</i>	20



5.7.10	Logout utente	20
5.7.11	Timeout di sessione	20
5.7.12	Isolamento delle funzioni dall'applicazione.....	20
5.8	PASSWORD, CHIAVI E CERTIFICATI	20
5.8.1	Gestione di password, chiavi e certificati.....	21
5.8.2	Trasmissione delle password in rete	21
5.8.3	Generazione/conservazione delle password nel filesystem/DB.....	21
5.8.4	Batch Job dell'applicazione	21
5.8.5	Storage dei dati applicativi	21
5.8.6	Integrità delle informazioni.....	21
5.8.7	Meccanismi di autenticazione.....	21
5.8.8	Non ripudio delle sessioni	21
5.8.9	Schemi di sicurezza e crittografici.....	21
5.8.10	Weak Keys e Collision	22
5.8.11	URL cifrati	22
5.8.12	Normalizzazione dei dati cifrati.....	22
6	PRINCIPALI VULNERABILITÀ DERIVANTI DA ERRORI DI PROGRAMMAZIONE: OVERVIEW	23
6.1	VALIDAZIONE DELL'INPUT.....	23
6.1.1	Shell Execution Command.....	23
6.1.2	File Inclusion.....	24
6.1.3	XML external entity (XXE) injection.....	25
6.1.4	Insecure Deserialization	26
6.1.5	Cross Site Scripting (XSS).....	26
6.1.6	Directory Traversal.....	27
6.1.7	SQL Injection	28
6.2	SESSION MANAGEMENT	29
6.2.1	Session Stealing e HjiHacking.....	29
6.2.1.1	Cookie.....	30
6.2.1.2	Token di sessione.....	31
6.2.1.3	Accesso ad aree non autorizzate	31
6.3	CRITTOGRAFIA	32
6.3.1	Sniffing e algoritmi crittografici deboli	32
6.3.2	Brute forcing	33
6.3.3	Rainbow table e salt value	34
6.3.4	Archiviazione insicura	34
6.4	GESTIONE DEGLI ERRORI, DELLE ECCEZIONI	35
6.4.1	User Enumeration	36
6.4.2	Information disclosure	36
6.4.3	Directory Listing	38
6.4.4	Denial of Service (DoS)	38
6.4.5	Race condition.....	39
6.4.6	Privilege Escalation e aggiramento dei permessi utente	40
6.5	BOUND CHECKING E PROBLEMATICHE DI OVERFLOW	40
6.5.1	Stack overflow.....	41
6.5.2	Off-by-one/Off-by-few	41
6.5.3	Format string overflow	42
6.5.4	Heap overflow.....	43
6.5.5	Integer overflow ed altri errori logici di programmazione.....	45
6.6	PROCESSI DI TRACCIAMENTO	45
6.6.1	Agevolazione delle attività malevole dell'aggressore.....	45
6.6.2	Oscuramento delle attività dell'aggressore	46
7	BEST PRACTICES PER LO SVILUPPO IN SICUREZZA	47
7.1	C/C++.....	47
7.1.1	Cross-site scripting (XSS).....	47
7.1.2	Command Injection	48
7.1.3	Connection String Injection	49



7.1.4	Resource Injection	51
7.1.5	SQL Injection	52
7.1.6	LDAP Injection	53
7.1.7	Process control	53
7.1.8	Ulteriori indicazioni per lo sviluppo sicuro	54
7.1.8.1	Dichiarazioni	54
7.1.8.2	Utilizzo dei tipi di dati	55
7.1.8.3	Bitfields	56
7.1.8.4	Macro	56
7.1.8.5	L'operatore sizeof e il passaggio di dati come parametri	57
7.1.8.6	Allocazione dinamica	57
7.1.8.7	Deallocazione	57
7.1.8.8	Puntatori	58
7.1.8.9	Casting e problematiche di gestione delle variabili numeriche	58
7.1.8.10	Computazione e condizionali	59
7.1.8.11	Controllo del flusso	59
7.1.8.12	Passaggio di argomenti	59
7.1.8.13	Valori di ritorno	59
7.1.8.14	Chiamate a funzioni	60
7.1.8.15	Files	60
7.1.8.16	Gestione degli errori	60
7.1.8.17	Sicurezza dell'applicazione	60
7.2	JAVA	60
7.2.1	Cross-site scripting (XSS)	60
7.2.2	Code injection	61
7.2.3	Command injection	62
7.2.4	Connection string injection	63
7.2.5	LDAP Injection	64
7.2.6	Resource Injection	65
7.2.7	SQL injection	66
7.2.8	XPath injection	66
7.2.9	XML External Entity (XXE) injection	68
7.2.10	Ulteriori indicazioni per lo sviluppo sicuro	68
7.2.10.1	Inizializzazione	69
7.2.10.2	Visibilità	70
7.2.10.3	Modificatori	70
7.2.10.4	Utilizzo degli oggetti mutevoli	70
7.2.10.5	Definizione delle classi	71
7.2.10.6	Codice e permessi speciali	71
7.2.10.7	Esecuzione dei comandi di sistema	71
7.2.10.8	Oggetti	72
7.2.10.9	Serializzazione e deserializzazione	72
7.2.10.10	Memorizzazione delle informazioni riservate	73
7.2.10.11	Packages	73
7.2.10.12	Gestione delle eccezioni	73
7.2.10.13	Java Servlet	75
7.3	PL/SQL	78
7.3.1	Cross-site scripting (XSS)	78
7.3.2	Resource Injection	79
7.3.3	SQL Injection	79
7.3.4	Ulteriori indicazioni per lo sviluppo sicuro	80
7.3.4.1	Posizionamento delle procedure PL/SQL	80
7.3.4.2	Tipologie di procedure vulnerabili	81
7.3.4.3	Filtraggio dei tipi di input iniettabile	81
7.3.4.4	Filtro dei caratteri potenzialmente dannosi	81
7.3.4.5	Direttive per Oracle	81
7.4	JAVASCRIPT	83
7.4.1	Cross Site Scripting (XSS)	83
7.4.2	Client DOM Code Injection	84
7.4.3	Client DOM Stored Code Injection	85



7.4.4	Client DOM Stored XSS	85
7.4.5	Client DOM XSS	87
7.5	PYTHON	87
7.5.1	Cross-site scripting (XSS)	87
7.5.2	Code Injection	88
7.5.3	Command Injection	89
7.5.4	Connection String Injection	90
7.5.5	LDAP Injection	91
7.5.6	Resource Injection	92
7.5.7	SQL Injection	92
7.5.8	XPath Injection	93
7.5.9	XML External Entity (XXE) injection	94
7.5.10	OS Access Violation	94
7.5.11	Unsecure deserialization	95
7.6	C#	96
7.6.1	Cross-site scripting (XSS)	96
7.6.2	Code Injection	97
7.6.3	Command Injection	98
7.6.4	Connection String Injection	99
7.6.5	LDAP Injection	101
7.6.6	Resource Injection	101
7.6.7	SQL Injection	102
7.6.8	XPath Injection	102
7.6.9	XML External Entity (XXE) injection	103
7.6.10	Ulteriori indicazioni per lo sviluppo sicuro	104
7.6.10.1	Managed Wrapper per l'implementazione del codice nativo	104
7.6.10.2	Library Code che espone risorse protette	104
7.6.10.3	Richieste di autorizzazione	104
7.6.10.4	Protezione dell'accesso ai metodi	105
7.6.10.5	Protezione e campi pubblici di sola lettura	106
7.6.10.6	Esclusione di classi e membri utilizzati da codice non attendibile	106
7.6.10.7	Definizione delle classi	108
7.6.10.8	User input	108
7.6.10.9	Concorrenza	108
7.6.10.10	Serializzazione e deserializzazione	109
7.7	ASP	109
7.7.1	Cross-site scripting (XSS)	109
7.7.2	Code Injection	111
7.7.3	Command Injection	111
7.7.4	Connection String Injection	112
7.7.5	LDAP Injection	113
7.7.6	XPath Injection	113
7.7.7	Resource Injection	114
7.7.8	SQL Injection	114
7.8	ASP.NET	115
7.8.1	Cross-site scripting (XSS)	115
7.8.2	Code Injection	116
7.8.3	Command Injection	117
7.8.4	Connection String Injection	118
7.8.5	LDAP Injection	119
7.8.6	Resource Injection	120
7.8.7	SQL Injection	120
7.8.8	XPath Injection	120
7.8.9	Ulteriori indicazioni per lo sviluppo sicuro	121
7.8.9.1	ASP.NET Web Form	121
7.8.9.2	ASP.NET MVC	122
7.9	PHP	123
7.9.1	Cross-site scripting (XSS)	123



7.9.2	Code Injection	124
7.9.3	Command Injection	126
7.9.4	File Disclosure	127
7.9.5	Remote File Inclusion	127
7.9.6	File Manipulation	128
7.9.7	LDAP Injection	129
7.9.8	Reflected Injection.....	130
7.9.9	SQL Injection	131
7.9.10	XPath Injection.....	131
7.9.11	XML External Entity (XXE) injection	132
7.9.12	Unsecure deserialization.....	133
7.10	VBNET.....	134
7.10.1	Cross-site scripting (XSS).....	134
7.10.2	Code Injection	135
7.10.3	Command Injection.....	136
7.10.4	Connection String Injection.....	136
7.10.5	LDAP Injection.....	137
7.10.6	Resource Injection.....	138
7.10.7	SQL Injection.....	138
7.10.8	XPath Injection.....	139
7.11	AJAX.....	139
7.11.1	Client Dom Code Injection	140
7.11.2	Client DOM Stored Code Injection	141
7.11.3	Client Dom Stored XSS	141
7.11.4	Client Dom XSS.....	143
7.11.5	Client Resource Injection	143
7.11.6	Client Second Order Sql Injection.....	144
7.11.7	Client Sql Injection	145
7.11.8	Cross-Site Request Forgery (CSRF).....	145
7.12	GO	147
7.12.1	Client Dom Stored XSS	147
7.12.2	SQL Injection	150
7.12.3	Ulteriori indicazioni per lo sviluppo sicuro	151
7.12.3.1	Validazione dell'INPUT.....	151
7.12.3.2	Gestione dei File	152
7.12.3.3	Gestione Sessione, Controlli Accessi e Crittografia.....	153
7.12.3.4	Gestione degli Errori e delle Eccezioni.....	155
7.12.3.5	Sicurezza del Database	156

LISTA DELLE TABELLE

Tabella 1 - Documenti di Riferimento.....	8
Tabella 2 - Definizioni	9
Tabella 3 - Acronimi	11

LISTA DELLE FIGURE

Figura 1 - Schema per la sicurezza dell'applicazione	12
--	----

1 INTRODUZIONE

1.1 Scopo

Scopo del presente documento è supportare, attraverso opportune linee guida, lo sviluppo di applicazioni software sicure. Queste linee guida, costituiscono un insieme di best practices da seguire, al fine prevenire eventuali problematiche di sicurezza nel codice, e forniscono nel contempo uno strumento utile nell'individuazione di possibili vulnerabilità presenti nel codice sorgente e le relative contromisure da applicare.

1.2 Struttura del documento

Il presente documento è articolato come segue:

- Il Capitolo 1 riporta le generalità e lo scopo del documento;
- Il Capitolo 2 riporta la documentazione applicabile e di riferimento al presente documento;
- Il Capitolo 3 riporta le definizioni e gli acronimi utili per la lettura del documento;
- Il Capitolo 4 riporta un'introduzione alle applicazioni sicure;
- Il Capitolo 5 fornisce un insieme di raccomandazioni generali e trasversali alle scelte implementative;
- Il Capitolo 6 fornisce un elenco delle principali vulnerabilità software, corredate da esempi puntuali e delle relative contromisure da adottare;
- Il Capitolo 7 fornisce le best practices per i linguaggi di sviluppo utilizzati (C/C++, Java, PL/SQL, Javascript, PyThon, C#, ASP, ASP.NET, PHP, VBNET, AJAX, GO) e delle misure da adottare al fine di diminuire l'esposizione verso problematiche di sicurezza applicativa.