CAPITOLO 19: COSTRUIRE SISTEMI IN SICUREZZA

(a.a. 2007/2008)

- Professore: ***Stefano Bistarelli***
- Studentessa: ***Assunta Di Giorgio***

… COSTRUIRE SISTEMI IN SICUREZZA …

- **Sicurezza nella definizione dei requisiti e nell'analisi**
- Sicurezza durante la progettazione del sistema
- Considerazioni di implementazione che supportano la sicurezza
- Sicurezza durante operazioni e manutenzione

SICUREZZA NELLA DEFINZIONE DEI REQUISITI E NELL'ANALISI

La sicurezza nello sviluppo di requisiti consiste nel capire quali requisiti essa deve fornire.

L'insieme dei requisiti deve essere completo e corretto.

Definire i requisiti è un processo iterativo che normalmente inizia con la *definizione di minaccia* e termina con i requisiti dettagliati per ogni fase, cioè che sono utilizzati nella progettazione, nell'implementazione e nella manutenzione del sistema.

MINACCE E OBIETTIVI DI SICUREZZA(1)

Nel costruire un sistema sicuro o affidabile, non bisogna credere che le minacce al sistema siano evidenti e ben definite.

Definizione: **Una minaccia è un evento potenziale che può avere un effetto indesiderato sul sistema e le sue risorse.**

Le *minacce alla sicurezza* sono:

- violazioni alla riservatezza
- distruzione dell'integrità
- negazioni di servizio.

MINACCE E OBIETTIVI DI SICUREZZA(2)

Ogni minaccia, una volta identificata, deve essere indirizzata da una qualche contromisura che cerca di contrastarla.

Una di queste contromisura consiste nello sviluppo dei requisiti di sicurezza di alto livello, chiamati anche ***obiettivi di sicurezza***, che servono proprio a tale scopo.

Essi forniscono indizi sui tipi dei meccanismi necessari ad implementarli e rivelano informazioni che possono aiutare nello sviluppo successivo di una specifica dettagliata di requisiti.

Un esempio di questi obiettivi richiede l'identificazione e l'autenticazione dell'utente prima che a quest'ultimo sia dato l'accesso a qualsiasi risorsa del sistema.

MINACCE E OBIETTIVI DI SICUREZZA(3)

Le minacce sono diverse dalle vulnerabilità.

Definizione: **Una vulnerabilità è una debolezza che rende possibile l'attuazione di una minaccia.**

Se un utente può accedere all'informazione di un'applicazione o di un database accedendo direttamente al sistema operativo, allora il sistema è vulnerabile.

È necessario distinguere le minacce in relazione al sistema specifico e all'ambiente in cui esso opera.

Le minacce possono provenire da fuori o dentro i confini che definiscono il sistema.

Se il sistema non è collegato a reti esterne, gli aggressori esterni potrebbero non costituire una minaccia.

MINACCE E OBIETTIVI DI SICUREZZA(4)

Di solito gli utenti interni sono persone fidate che utilizzano il sistema correttamente, ma ci sono molti modi in cui questa correttezza può venir meno.

Una di queste modalità consiste nell'**uso improprio intenzionale delle autorizzazioni**.

Quindi le minacce possono provenire:

- da utenti autorizzati oppure
- utenti non autorizzati che
  - si mascherano come utenti validi oppure
  - trovano altri metodi per saltare i meccanismi di sicurezza e raggiungono le informazioni che comunemente sarebbero loro negate da parte dell'autore.

Possono anche essere causati da errori umani o pure casualità.

Ad esempio c'è l'*ERRORE FAT-FINGER*, dove l'utente autorizzato commette un errore, usando impropriamente o alterando involontariamente il sistema.

CONSIDERAZIONI SULL'ARCHITETTURA DEL SISTEMA

1- Una decisione riguardante l'architettura è quella di *determinare il centro di controllo dei meccanismi di rafforzamento della sicurezza*.

Esempio: la ***sicurezza dei computer si basa sull'accesso alle informazioni***, che può essere filtrato tramite meccanismi di sicurezza e operazioni che impongono all'utente di identificarsi e autenticarsi. In questo caso si controllano i privilegi attribuitigli, in modo tale che egli possa svolgere solo operazioni consentite.

2- Un'altra decisione consiste nella scelta di *centralizzare alcune funzioni di sicurezza in un unico modulo* oppure distribuirle in più moduli presenti nel sistema.

Generalmente è più facile sviluppare e analizzare sistemi centralizzati, in modo tale che ***l'analisi venga svolta una sola volta e in un unico posto***, mentre le altre fasi della sicurezza siano affidate a routines che vengono chiamate appropriatamente.

MECCANISMI DI SICUREZZA E ARCHITETTURA STRATIFICATA

- Le architetture dei computer sono *stratificate* e i meccanismi di rafforzamento della sicurezza possono risiedere in alcuni livelli.
- I sistemi disegnati e costruiti a tal fine descrivono con precisione la funzionalità di ogni livello.

PROCESSO DI UNA RICHIESTA

Quando un'applicazione riceve una richiesta, passa la richiesta al livello presente sotto l'applicazione (cioè il *livello applicativo*).

Quel livello processa la richiesta e la passa al livello successivo (il *livello intermedio o di servizio*).

Questo processo continua fino a quando la richiesta non raggiunge il livello che può soddisfarla.

I livelli successivi seguono semplicemente le istruzioni che vengono date dai livelli precedenti.

Quando la richiesta viene soddisfatta, l'informazione corrispondente viene passata all'utente a livello applicativo.

UN MECCANISMO DI SICUREZZA ALL'INTERNO DI UN LIVELLO

3- Un'ulteriore decisione sull'architettura riguarda la *selezione del livello corretto per un determinato meccanismo*.

I disegnatori devono selezionare il livello a cui il meccanismo sarà più efficiente e più efficace.

I meccanismi di sicurezza per il controllo delle ***azioni dell'utente*** possono essere ***più efficaci a livello applicativo***, ma i meccanismi di sicurezza per la *cancellazione dei dati* possono essere *più efficaci al livello di sistema operativo*.

Una volta che un livello viene scelto per accogliere un determinato meccanismo di sicurezza, bisogna vedere come proteggere il livello al di sotto del livello in questione.

Potrebbe non essere possibile piazzare un meccanismo nel livello scelto: il meccanismo potrebbe essere piazzato in un livello meno ottimale oppure i costruttori devono far sì che il livello venga specializzato.

COSTRUIRE SICUREZZA O AGGIUNGERLA IN UN SECONDO MOMENTO

Se costruiamo un prodotto ad alte prestazioni a partire da uno dalle prestazioni povere ed esse sono attribuibili a funzioni specifiche, queste ultime devono essere riprogettate.

Correggere la struttura base e la progettazione del sistema è un problema molto difficile e il sistema ottenuto non presenta la stessa affidabilità di uno che è stato realizzato soddisfando dall'inizio tutti i requisiti di sicurezza.

Alla base della progettazione e dello sviluppo di sistemi computerizzati sicuri è necessario avere:

1. un **monitor di riferimento (Reference Monitor)**: è il concetto di una macchina astratta che applica e fa rispettare le politiche di controllo d'accesso.
2. un **meccanismo di validazione di riferimento (Reference Validation Mechanism)**: è l'implementazione del concetto di reference monitor .

Le **caratteristiche di un RVM** sono:

- deve essere a prova di intrusione (nessuna vulnerabilità),
- deve essere sempre richiamato (e mai scavalcato), quindi invocato ad ogni richiesta di accesso e
- deve essere abbastanza semplice da essere verificabile, cioè sottoposto ad analisi e verifiche minuziose, assicurandone la completezza.

TCB(1)

Un **TCB (Trusted Computing Base)** è definito dall'insieme di tutti i meccanismi di protezione del sistema (hardware, componenti software, firmware) la cui combinazione è responsabile per l'attuazione della politica di sicurezza.

Questa sua capacità dipende esclusivamente:

- dai meccanismi presenti al suo interno e
- dall'immissione corretta dei parametri riferibili alla politica di sicurezza.

L'attenta progettazione e attuazione di un sistema di TCB è fondamentale per la sua sicurezza globale.

TCB(2)

I sistemi in cui i meccanismi di sicurezza vengono aggiunti in un secondo momento non possono essere sottoposti alla stessa analisi approfondita fatta per gli altri sistemi.

Un'analisi rigorosa di sistemi complessi è difficile, quindi potrebbe non essere fattibile determinare come la progettazione implementa i requisiti.

In teoria il divario (o gap) tra i requisiti di sicurezza e il codice di implementazione potrebbe impedire la verifica completa dei requisiti.

Un esempio di sistema operativo realizzato col fine principale della sicurezza è ***MULTICS (Multiplexed Information and Computing Service)***, che mise in campo tutta una serie di concetti e tecniche costruttive che sono ancora oggi elementi essenziali dei moderni sistemi operativi.

DEFINIZIONE DELLA POLITICA E SPECIFICHE DI REQUISITI(1)

Definizione: **Una specifica è una descrizione delle caratteristiche di un computer system o di un programma. Una specifica di sicurezza descrive appunto le proprietà desiderate di sicurezza.**

Le specifiche sono importanti tanto quanto le proprietà dei sistemi che vanno a descrivere. Devono essere ***chiare, non ambigue e complete***, anche se è difficile da ottenere quando si usano metodi informali che possono contare sul linguaggio naturale che non ha sintassi e semantica precise.

Può essere difficile ottenere precisione anche nella definizione dei requisiti.

I requisiti standard sono:

1. *Gli utenti del sistema devono essere identificati e autenticati*: requisito ambiguo.
2. *Gli utenti del sistema devono essere identificati dal sistema e devono fare in modo tale che l'identificazione sia autenticata dal sistema stesso*: requisito più preciso.
3. *Gli utenti del sistema devono essere identificati dal sistema e devono fare in modo tale che l'identificazione sia autenticata dal sistema stesso prima che essi svolgano qualche funzione per conto di tali identità*: requisito ottimo.

DEFINIZIONE DELLA POLITICA E SPECIFICHE DI REQUISITI(2)

Ci sono molti metodi per definire le politiche o le specifiche dei requisiti.

1. Estrarre requisiti applicabili dagli standard di sicurezza esistenti. Queste specifiche tendono ad essere semiformali a causa della struttura dei requisiti e delle corrispondenze tra loro.
2. Creare una nuova politica combinando i risultati delle analisi di una minaccia con i componenti delle politiche esistenti.
3. Mappare un sistema rispetto ad un altro. Se il modello è appropriato per gli obiettivi del sistema, creare un confronto tra il modello e il sistema potrebbe risultare più semplice e più economico rispetto al costruire una specifica di requisiti a partire da altri metodi.
   - Se il confronto è accurato, le prove del modello originale stabiliscono la correttezza della politica in questione.

GIUSTIFICARE I REQUISITI

Una volta che la politica viene definita e specificata, deve essere completa e coerente con le altre politiche.

L'ITSEC è un'armonizzazione dei criteri di valutazione di molti Paesi europei. Ha introdotto il concetto di "obiettivo di sicurezza" che definisce le minacce alla sicurezza al sistema e i requisiti funzionali del sistema posto a valutazione.

Un'analisi di idoneità dell'ITSEC verifica se i requisiti funzionali di sicurezza sono sufficienti a far fronte alle minacce al sistema.

I requisiti e le ipotesi rappresentano il SECURITY TARGET REFERENCE, cioè controllano l'accesso non autorizzato.

Se le ipotesi sono ben formulate e i requisiti sono adeguati, la minaccia viene adeguatamente gestita.

… COSTRUIRE SISTEMI IN SICUREZZA …

- Sicurezza nella definizione dei requisiti e nell'analisi
- **Sicurezza durante la progettazione del sistema**
- Considerazioni di implementazione che supportano la sicurezza
- Sicurezza durante operazioni e manutenzione

SICUREZZA DURANTE LA PROGETTAZIONE DEL SISTEMA

La sicurezza nella progettazione viene spesso trascurata. I difetti di progettazione vengono di solito scoperti quando i test evidenziano anomalie che difficilmente possono essere corrette.

Se la progettazione fosse analizzata, i difetti nella sicurezza potrebbero essere corretti in questa fase, mentre quelli nella realizzazione sono più facili da risistemare.

La progettazione della sicurezza è il processo per stabilire che la progettazione del sistema è sufficiente a rafforzare i requisiti di sicurezza del sistema stesso.

Le ***tecniche di progettazione della sicurezza*** si basano su:

- la specifica dei requisiti,
- la specifica della progettazione del sistema e
- i processi che portano ad esaminare quanto la progettazione soddisfa i requisiti.

LE TECNICHE DI PROGETTAZIONE CHE SUPPORTANO LA SICUREZZA

La modularità e la stratificazione sono tecniche della progettazione e dell'implementazione del sistema che possono semplificare il sistema, rendendolo così più disposto ad essere sottoposto ad analisi di sicurezza.

Il meccanismo RVM suggerisce che le funzioni non relative alla sicurezza siano rimosse dai moduli che supportano le funzionalità di sicurezza. Questo fa sì che ***i moduli siano più piccoli e più facili da analizzare***. Questi concetti di progettazione devono essere attentamente descritti nella documentazione di progettazione e di implementazione.

Definizione: **Un sottosistema o componente è una divisione specializzata di un'entità più grande.**

Definizione: *Un componente è costituito da strutture di dati e sottocomponenti o moduli.*

Definizione: *Un modulo è un insieme di funzioni collegate tra loro e strutture dati pertinenti.*

CONTENUTO DEL DOCUMENTO DI PROGETTAZIONE

La maggior parte dei prodotti/sistemi richiede la documentazione di progettazione, benchè i requisiti della documentazione non siano sempre sufficienti per sviluppare la sicurezza nella progettazione.

Una specifica più rigorosa può essere necessaria per stabilire che la progettazione del sistema sia sufficiente a rafforzare i requisiti di sicurezza.

Le specifiche di progettazione possono essere informali, semiformali o anche formali.

Le specifiche che sono più formali possono essere soggette ad analisi e giustificazioni più rigorose, fornendo un alto livello di sicurezza.

Un vantaggio significativo nello scrivere le specifiche consiste nell'abilità di correggere una progettazione grazie alla sua definizione messa per iscritto.

Più precise sono le descrizioni, più risulterà facile individuare gli errori e correggerli.

Per l'analisi di sicurezza, la documentazione deve specificare tre tipi di informazione:

1. FUNZIONI DI SICUREZZA
2. INTERFACCE ESTERNE
3. PROGETTAZIONE INTERNA.

LA SPECIFICA RIEPILOGATIVA DELLE FUNZIONI DI SICUREZZA

È il più alto livello di specifica del rafforzamento della sicurezza ed è significativo per lo sviluppo di tutte le specifiche successive e per l'analisi di sicurezza da cui essi dipendono.

Una specifica riepilogativa delle funzioni di sicurezza identifica le funzioni di sicurezza a più alto livello che sono definite per il sistema, cioè ***SPECIFICA AD ALTO LIVELLO IMPLICA MAGGIORE SICUREZZA***.

Il contenuto di queste funzioni dovrebbe includere le seguenti informazioni:

- DESCRIZIONE DELLE FUNZIONI INDIVIDUALI DI SICUREZZA
- PANORAMICA DELL'INSIEME DELLE FUNZIONI DI SICUREZZA
- MAPPATURA DEI REQUISITI.

SPECIFICA FUNZIONALE ESTERNA

Le descrizioni di ogni interfaccia esterna forniscono dettagli su parametri, effetti e condizioni di errore.

Ogni funzione di sicurezza può avere numerose interfacce visibili agli utenti, che sono di particolare importanza per una specifica di un sistema oppure un prodotto sicuro o affidabile.

Una *specifica funzionale esterna*, chiamata anche ***specifica funzionale***:

- è una descrizione ad alto livello delle interfacce esterne verso un sistema, un componente, un sottocomponente o un modulo;
- può essere scritta per un intero sistema, un componente, un sottocomponente o anche un modulo.

Il contenuto tecnico di questa specifica dovrebbe includere le seguenti informazioni:

- PANORAMICA DEI COMPONENTI
- DESCRIZIONE DEI DATI
- DESCRIZIONE DELL'INTERFACCIA.

DESCRIZIONE DELLA PROGETTAZIONE INTERNA

Una descrizione della progettazione interna *descrive appunto le strutture interne e le funzioni dei componenti del sistema*.

Questa descrizione consiste in uno o più documenti.

La complessità del sistema e la sua suddivisione in componenti e sottocomponenti determina la suddivisione della documentazione:

- di alto livello e
- di basso livello della progettazione.

DOCUMENTI DI ALTO LIVELLO DELLA PROGETTAZIONE

I documenti di alto livello *forniscono informazioni specifiche sulla progettazione del sistema, in termini di componenti e sottocomponenti, indipendentemente dal livello di suddivisione della progettazione*.

Il loro contenuto tecnico include le seguenti informazioni:

- PANORAMICA SUL COMPONENTE SUPERIORE
- DESCRIZIONE DETTAGLIATA DEL COMPONENTE
- RILEVANZA DELLA SICUREZZA DEL COMPONENTE.

DOCUMENTI DI BASSO LIVELLO DELLA PROGETTAZIONE

I documenti di basso livello si focalizzano sulla progettazione interna di moduli, descrivendo le strutture dati rilevanti e interfacce.

Includono descrizioni dettagliate delle funzioni delle interfacce.

La specifica analizza come una funzione viene implementata e potrebbe includere specifici algoritmi e pseudocodici.

Una descrizione di basso livello di un modulo *dovrebbe contenere informazioni sufficienti ad uno sviluppatore per scrivere il codice di implementazione del modulo*.

Contiene le seguenti informazioni:

- PANORAMICA DEL MODULO CHE DEVE ESSERE SPECIFICATO
- RILEVANZA DELLA SICUREZZA DEL MODULO
- INTERFACCE INDIVIDUALI DEL MODULO.

SPECIFICA DI PROGETTAZIONE INTERNA

Questa è leggermente più complessa rispetto alle precedenti.

Gli sviluppatori potrebbero usare un documento di specifica di progettazione interna che *copre le parti dei documenti sia di alto che di basso livello*.

È un insieme di documenti utile, leggibile e completo.

È la più usata quando vengono specificati:

- i livelli di suddivisione di un sistema e
- i moduli che si trovano in quei livelli.

COSTRUIRE DOCUMENTAZIONE E SPECIFICHE

Il tempo, il costo e gli aspetti di efficienza possono influire su come un'organizzazione di sviluppo crea un insieme completo di documenti.

Esempio: un vincolo di tempo può costringere un'organizzazione a scrivere specifiche informali piuttosto che formali.

SPECIFICHE DI MODIFICA

Quando un sistema o un prodotto è costruito a partire da versioni precedenti o componenti esistenti, l'insieme di specifiche può essere formato da specifiche delle precedenti versioni con le specifiche di modifica che descrivono i cambiamenti che sono stati apportati.

Le *specifiche di modifica* descrivono:

- i cambiamenti nei moduli, funzioni o componenti esistenti;
- l'aggiunta di nuovi moduli, funzioni o componenti;
- i metodi per cancellare moduli, funzioni o componenti scartati.

L'analisi di sicurezza si deve basare sulla specifica del prodotto risultante, non solo i cambiamenti apportati, altrimenti l'analisi risulterebbe incompleta.

I problemi aumentano quando le specifiche di modifica sono le sole specifiche del sistema.

SPECIFICHE DI SICUREZZA

Quando le specifiche interne ed esterne di progettazione sono adeguate sotto tutti i punti di vista, tranne che per gli aspetti di sicurezza, *una specifica supplementare può essere creata per descrivere la funzionalità che non c'è.*

Un approccio consiste nella redazione di un documento che parte dalla specifica riepilogativa delle funzioni di sicurezza. Viene esteso poi per indirizzare gli aspetti di sicurezza dei componenti, sottocomponenti, moduli e funzioni.

DIMOSTRARE CHE LA PROGETTAZIONE SODDISFA I REQUISITI

La natura (formale, informale, semiformale) della specifica limita le tecniche che possono convalidare la progettazione.

Le specifiche informali e quelle semiformali non possono essere analizzate usando metodi formali a causa dell'imprecisione del linguaggio da loro utilizzato.

Una tecnica eccellente per verificare alcune tecniche informali è chiamata ***REVIEW*** (revisione).

Altri metodi, che producono maggiore sicurezza, sono formali per natura.

REVIEW

Un meccanismo per valutare l'adeguatezza di determinate caratteristiche riguardo la sicurezza risulta più importante quando la tecnica utilizzata ha natura informale.

Ogni processo significativo di *review* ha tre parti critiche:

1. REVISIONE DELLE DIRETTIVE
2. METODI DI RISOLUZIONE DEI CONFLITTI
3. PROCEDURE DI COMPLETAMENTO.

I revisori ricevono (o meglio determinano) le direttive su come revisionare un'entità. Queste direttive vanno dalle più generali a quelle più specifiche.

I revisori possono avere opinioni ed esperienze diverse. ***Il processo di revisione deve avere un metodo per risolvere i conflitti tra revisori e autori.***

Infine, la revisione deve terminare, assicurando la completezza dell'entità che viene revisionata.

IL TRACING DEI REQUISITI E LA CORRISPONDENZA INFORMALE

Ci sono due tecniche che aiutano ad impedire che requisiti e funzionalità siano scartati, dimenticati o ignorati ai livelli più bassi della progettazione. Evidenziano anche le funzionalità che possono entrare nella progettazione ma non soddisfano i requisiti.

Queste tecniche sono:

1. Il ***TRACING DEI REQUISITI***: è il processo di identificazione dei requisiti di sicurezza che soddisfano solo parte dei requisiti specifici.

2. La ***CORRISPONDENZA INFORMALE***: è il processo che consiste nel mostrare che una specifica è coerente col livello adiacente di specifica.

Insieme, questi due metodi *riescono a fornire affidabilità data dalle specifiche che costituiscono una realizzazione completa e coerente dei requisiti di sicurezza definiti per il sistema*.

… COSTRUIRE SISTEMI IN SICUREZZA …

- Sicurezza nella definizione dei requisiti e nell'analisi
- Sicurezza durante la progettazione del sistema
- **Considerazioni di implementazione che supportano la sicurezza**
- Sicurezza durante operazioni e manutenzione

CONSIDERAZIONI DI IMPLEMENTAZIONE CHE SUPPORTANO LA SICUREZZA

Alcuni linguaggi supportano bene la sicurezza fornendo caratteristiche incorporate che aiutano ad evitare difetti ricorrenti. I programmi scritti in questi linguaggi sono spesso più affidabili.

I linguaggi che forniscono caratteristiche che supportano la sicurezza rivelano molti errori di realizzazione.

I linguaggi, aventi caratteristiche tipo modularità, protezioni di accesso, garbage collection e gestione degli errori, supportano lo sviluppo di programmi più sicuri, attendibili e affidabili.

Talvolta non è fattibile usare un linguaggio di alto livello a causa di vincoli di efficienza o della necessità di sfruttare le caratteristiche del sistema a cui il linguaggio ad alto livello non può accedere.

SICUREZZA ATTRAVERSO LA GESTIONE DELL'IMPLEMENTAZIONE

Gruppi di programmatori spesso sviluppano sistemi progettati in moduli. Ogni programmatore sviluppa i moduli indipendentemente dagli altri. Le interfacce modulo ben definite sono critiche.

La gestione della configurazione è il controllo dei cambiamenti avvenuti a livello di hardware e di software.

Il sistema di gestione della configurazione è composto da numerosi strumenti o processi manuali e dovrebbero eseguire molte funzioni:

- CONTROLLO E RILEVAZIONE DELLA VERSIONE
- CAMBIARE AUTORIZZAZIONI
- PROCEDURE DI INTEGRAZIONE
- STRUMENTI PER LA GENERAZIONE DI PRODOTTO.

Lo *sviluppo dei livelli di codice* è un altro strumento di gestione dell'implementazione che supporta la sicurezza.

NB: Nessun linguaggio di programmazione risolve tutti i problemi di sicurezza.

VERIFICHE DI SICUREZZA(1)

Ci sono due tipi di tecniche di verifica:

- ***VERIFICA FUNZIONALE***: talvolta chiamata “verifica della scatola nera”, è la verifica di un’entità per determinare come essa soddisfa le specifiche.
- ***VERIFICA STRUTTURALE***: talvolta chiamata “verifica della scatola bianca”, è una verifica basata su un’analisi del codice per poter sviluppare delle prove.

La verifica viene fatta più volte durante il processo di ingegnerizzazione:

1. *VERIFICA DI UNITA’*: consiste nella verifica da parte dello sviluppatore di un modulo di codice prima dell’integrazione. Di solito si tratta di una verifica *strutturale*.

2. *VERIFICA DEL SISTEMA*: è una verifica *funzionale* eseguita dal gruppo di integrazione sui moduli integrati del sistema. *Può includere in alcuni casi verifiche strutturali*.

VERIFICHE DI SICUREZZA(2)

3. *VERIFICA DA PARTE DI TERZI*: talvolta chiamata “verifica indipendente”, è una verifica eseguita da un gruppo che è fuori dall’organizzazione di sviluppo, spesso una società esterna.

4. *VERIFICA DI SICUREZZA*: è la verifica che indirizza la sicurezza del prodotto. Essa è composta da 3 componenti:
   - VERIFICA FUNZIONALE DI SICUREZZA: è la verifica funzionale specifica degli aspetti di sicurezza descritti nella specifica in oggetto.
   - VERIFICA STRUTTURALE DI SICUREZZA: è la verifica strutturale specifica dell’implementazione della sicurezza trovata nel codice in oggetto.
   - VERIFICA DEI REQUISITI DI SICUREZZA: è la verifica *funzionale* di sicurezza specifica dei requisiti di sicurezza presenti nella specifica dei requisiti. *Può sovrapporsi in modo significativo alla verifica funzionale di sicurezza*.

… COSTRUIRE SISTEMI IN SICUREZZA …

- Sicurezza nella definizione dei requisiti e nell'analisi
- Sicurezza durante la progettazione del sistema
- Considerazioni di implementazione che supportano la sicurezza
- **Sicurezza durante le operazioni e la manutenzione**

SICUREZZA DURANTE LE OPERAZIONI E LA MANUTENZIONE(1)

Mentre un sistema è in funzione, i bugs possono entrare, richiedendo manutenzione sul sistema.

Un **HOT FIX** è un pacchetto che contiene uno o più files che vengono utilizzati per indirizzare subito i ***bugs*** (chiamati comunemente *buchi nella sicurezza*) che vengono mandati fuori prima possibile, in quanto possono attaccare la sicurezza del sistema in maniera più grave.

Una **REGULAR FIX** indirizza i bugs meno gravi o fornisce soluzioni a lungo termine per i bugs già indirizzati con gli hot fixes.

SICUREZZA DURANTE LE OPERAZIONI E LA MANUTENZIONE(2)

Per quanto riguarda la manutenzione del sistema, le procedure ben definite si occupano dei difetti rilevati.

L'informazione su ogni difetto dovrebbe includere:

- una descrizione del difetto,
- le azioni correttive apportate o pianificate,
- la gravità e la priorità del difetto,
- ciò che dovrebbe essere modificato nel codice e nella documentazione
- etc.

L'azione intrapresa per un intervento di manutenzione o una correzione di bugs dovrebbe seguire le stesse procedure di sicurezza usate durante lo sviluppo originale.

FINE

BUONE FESTE