

PROSECUZIONE SLIDES:

Domanda Esame Esempio:

"Prendere Esempio Autoricarica Ogni Secondo"

"È un Buon Esempio? NO, ricaricate Troppo Spesso, si utilizza i Web Socket"

ORALE \Rightarrow Si Parte dagli Esempi Semplici ma Vanno fatti

BROWSER Solitamente non Apre file tramite OPEN ma di Solito li Trova Nella CONNESSIONE http con il SERVER.

Competenza Utile \Rightarrow Capire che Cosa gira Sul FRONT-END o BACK-END per Capire dove Collegare una Nuova Tecnologia.

Può Esserci Anche un MIDDLEWARE Tra i 2

DOM \Rightarrow document Object Model

HTML+Javascript \Rightarrow Lo Rende Dinamico, Viene Eseguito dal Browser e Rende Vivo l'HTML.

Tramite **Javascript** posso far fare Cose al BROWSER, anche Modificare La Stessa pagina

Non è più Quindi **RICHIESTA \rightarrow RISPOSTA**... Ma c'è il Concetto di **WEB-APPLICATION**, **WEB DINAMICO** (una Vera e Propria Appl.)

WEB STATICO (risposte Solo con Richieste)

Ad Esempio Google Docs rispetto a Microsoft Word
Web Application Software Stand Alone.

Oggetto **DOCUMENT** c'è Sempre non Serve istanziarlo

Get ElementById \Rightarrow Recuperò un Elemento Specifico

InnerHTML += "..." \Rightarrow Aggiungo dinamicamente il Testo
"..." a Quello PRE-ESISTENTE

SetInterval(n) \Rightarrow Eventi Sincroni

non più Stile Programmazione **BATCH**

sequenziale dall' Inizio alla fine

Processo finisce

WEB-APP Sono **EVENT DRIVEN**

Eventi

Serie di funzioni per Gestire gli

Non Esiste un ORDINE... Ovviamente
Solo All' Interno delle funzioni

■ EVENTI Sincroni \Rightarrow So già Quando Capita

Ad Esempio degli Allarmi \Rightarrow Set Interval (What, N)
Lo chiamo io Quindi so già Quando e Cosa fa

Ad Esempio Se Ricarica Periodicamente una Pagine Web,
questo si Chiama POLLING \Rightarrow Poco piacevole, soprattutto se
l'intervallo è Basso oltre ad Essere poco Efficiente

■ EVENTI A-SINCRONI \Rightarrow Non è Legato al TEMPO, non
So quando Capita (come un MESSAGGIO in Chat)

http NON permette che il Server in Maniera Asincrona
mandi INFO al Client e si Sono Inventati un PROTOCOLLO
di livello Applicativo \Rightarrow Web-Socket per Realizzarlo

CONCETTO DI URL:

Come Identificare una Risorsa in RETE (a livello Appl.)
altrimenti Come al Solito:

- IP \rightarrow lvl 3
- MAC \rightarrow Rete Locali

Modello ISO/OSI fornisce un CONCETTO di INDIRIZZO ad ogni
lvl (Tranne quello fisico)

- URL
- Porte (lvl 4)
- IP (lvl 3)
- MAC (lvl 2)

STRUTTURA

- m Protocollo \Rightarrow HTTPS (dal Quale capisco la PORTA STANDARD)
- m Nome dell' host \Rightarrow "www....it" (controparte dell' IP con DNS)
- m Nome della RISORSA sull' HOST \Rightarrow Percorso Logico Completo e Non Necessariamente FISICO (Come ERA una Volta)
 - Rimane un FILESYSTEM Gerarchico

SICUREZZA:

Un ABC Molto Semplice (PDF Per Approfondire)

Proteggersi Da delle Minacce....

- m Se Nascono dalla Natura \Rightarrow Affidabilità, come Fulmini / Alluvioni / Cavi di Rete Schiacciati / ...
- m Se la Causa è di TIPO Umano (ma NON errore, altim. ancora nell' Affidabilità) e se c'è il DOLORO allora è SECURITY

TCP non è Sicuro ma AFFIDABILE Mentre TLS/SSL è un protocollo per la SICUREZZA

fault / failure in SECURITY si chiama MINACCIA, e c'è dell' Intelligenza Umana Dietro

COSA SI VUOLE PROTEGGERE

Hardware, Software, Dati e la Rete.

Garentire Sicurezza: (3 proprietà)

ASPECTI CONCRETI
DELLA SICUREZZA

■ **Confidenzialità** ⇒ Segretezza

■ **Integrità** ⇒ Più facile de falsificare Rispetto a Qualcosa di fisico (DATO DIGITALE PIÙ SEMPLICE)

■ **Disponibilità** ⇒ Blocca l'Accesso al Servizio, che è un Danno fatale (DENY OF SERVICE).

Ad Esempio CIFRARE HDD e chiedere un riscatto. [non rubo nulla ma è dannoso]

■ **Autenticità** ⇒ Devo Riuscire a RICONOSCERE chi fa Cosa ed avere la Certezza (UN BEL PROBLEMA)

■ **Tracciabilità** ⇒ Tracciare e LOGGARE tutto Quello che è Successo per RISALIRE a chi ha fatto Cosa ⇒ RECUPERO DELLA STORIA PASSATA

5 Aspetti Sono ORTOGONALI, Non Sempre Sono Importanti TUTTI da Garentire (come cattivo della Cosa Esempio)

Siamo DIPENDENTI dell'ELETTRICITÀ ma Anche le RETE ELETTRICA è Totalmente INFORMATIZZATA.

Anche L'INDUSTRIA 4.0., ci Sono Possibilità di fare DANNI Enormi.

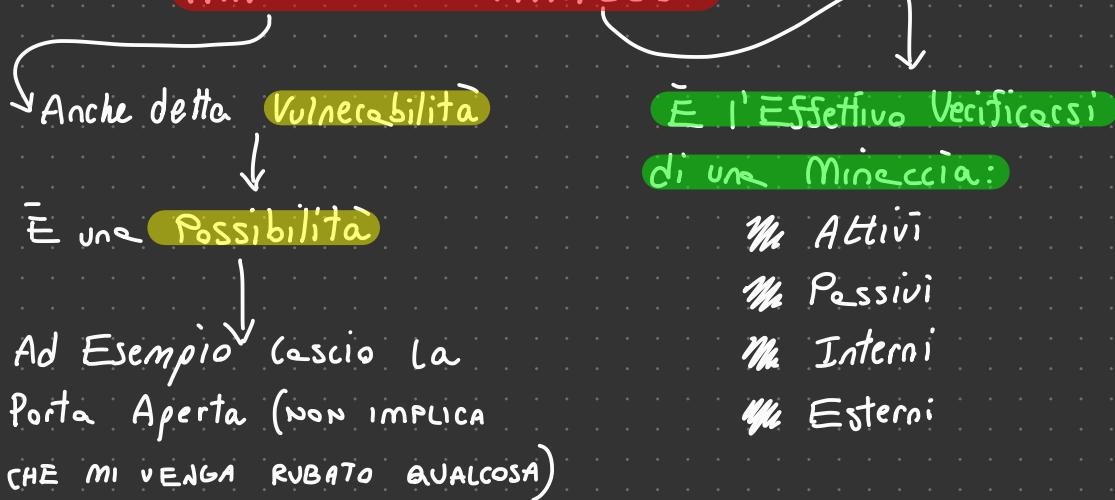
CONFIDENZIALITÀ/SEGRETEZZA:

Privacy è una SOTTOCATEGORIA ma è Ben diverso dalla CONFIDENZIALITÀ.

Concetto di **PRIVACY** significa che io Sono il PROPRIETARIO dei DATI. (Rimango in proprietà dei DATI e NON Vengono dati a Terzi)



MINACCIA o ATTACCO:



CyberSecurity è Legata Anche Alla Psicologia

Anche dall' Interno possono Arrivare gli Attacchi... Bisogna avere criticci di PROTEZIONE Anche dall' Interno, ad Esempio:

Un Amministratore di Sistema con Tutte le password, anche se Moralmente è Apposto ha un PUNTO da Attaccare del quale ho Accesso a TUTTO

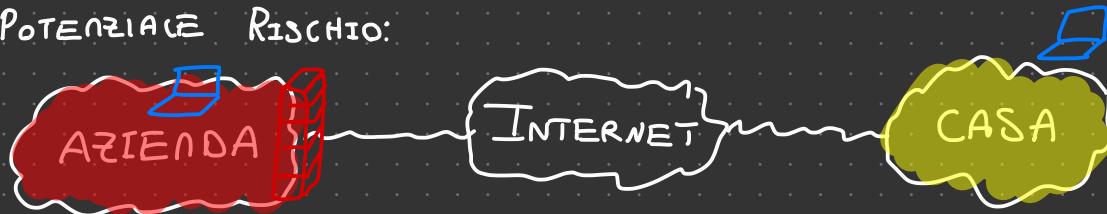
Devo Riuscire a Decentralizzare, altrimenti CREO una Vulnerabilità INTERNA.

Bisogna CREARE un Sistema il più MODULARE Possibile

La COMPLESSITÀ è un gran problema... ma facendoli a pezzi (MODULARITÀ - divid et Impera) e mantenendo

SISTEMA può Essere Sicurissimo. Ma devo FORMARE il personale per non Creare Vulnerabilità IRRISOLVIBILI come se un UTENTE Condivide le Sue PSW con il Suo COLLEGA.

Ma anche Avere PC Aziendale e Portarlo a Casa è un POTENZIALE RISCHIO:



Azienda con FIREWALL ed è Super Sicuro, ma Se SPOSTO la mia Macchina Nella RETE AZIENDALE il firewall non serve a Nulla (io posso dalla PORTA e non dal FIREWALL)

Se NON Coinvolgo Anche l'Aspetto Umano Allora il lavoro diventa INUTILE

C'è un Problema PSICOLOGICO alle Base

SICUREZZA non ha a che Vedere con le SEGRETEZZA dei Meccanismi, ad Esempio non dire il TIPO dell'Algoritmo che USO o l'ANTIVIRUS (magari NUOVO, allora aumenta il rischio)

SISTEMA OPEN SOURCE è Solitamente Migliore... più persone ci hanno lavorato Meglio è rispetto ad un Qualcosa di creato all' Interno)

Algoritmo di Cifratura può Essere NOTO / Deve Esserlo... ma non la chiave. (Se l'Algoritmo è NOTO e FUNZIONA BENE allora Senza chiave non deve permettere l'Accesso)

POLITICA D. SICUREZZA \Rightarrow Bloccare USB da BIOS ad Esempio