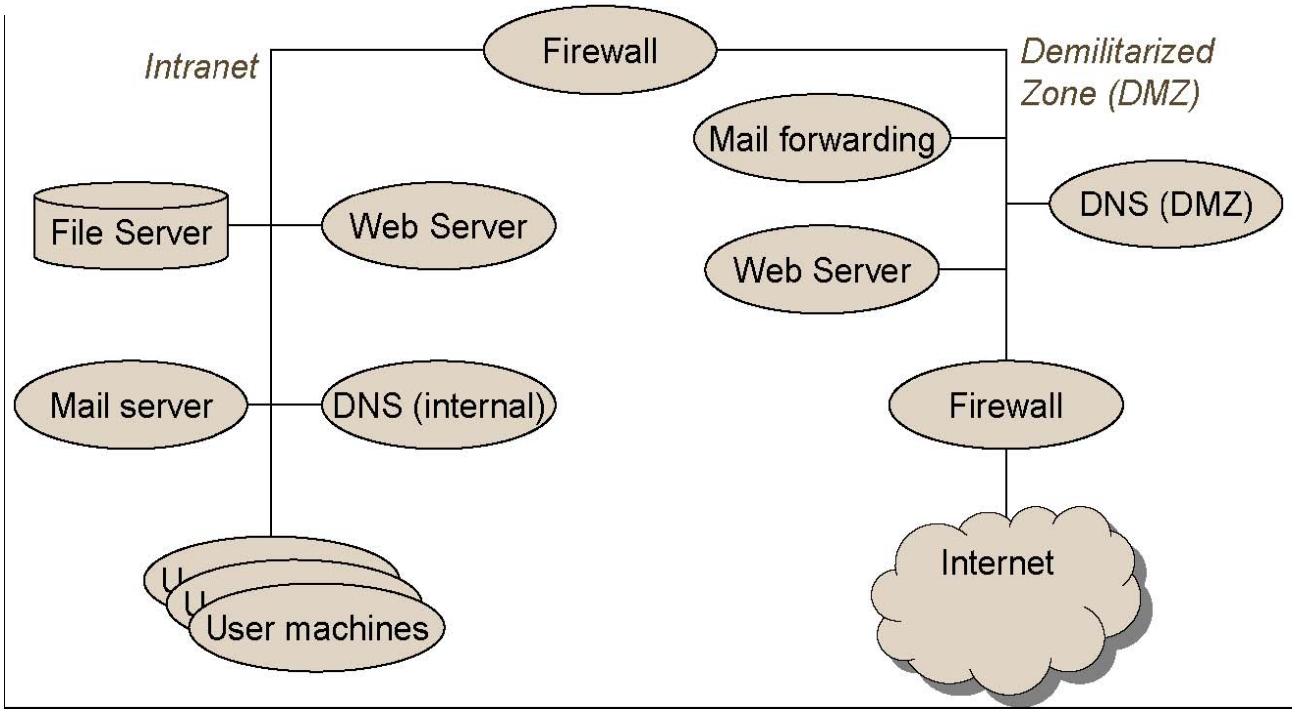
Corso di **Sicurezza A.A.2016-2017**
ESONERO aprile 2017

Le soluzioni devono essere ricevute entro le **22** di **martedì 2 maggio 2017,** in un file unico in **formato pdf : altri formati non sono accettabili.** La prima pagina delle soluzioni sottomesse deve riportare, in aggiunta al **Nome**, **Cognome** e **numero di matricola**, la seguente dichiarazione: **"Dichiaro di aver completato questo esonero da solo/a e di non aver discusso le soluzioni con altri studenti in classe."**

PROBLEMA 1 [15 punti]
Discutere quali degli otto principi di progettazione per sistemi sicuri sono rispettati e quali sono violati nella seguente tipica architettura di rete locale (per esempio, del Dipartimento di Informatica).



PROBLEMA 2 [5+5+10 punti]
Dato l'insieme di permessi {*own, execute, read, write, append, list, modify*}

1. Usando la sintassi HRU, scrivere un comando **delete_rights(s1,s2,f)** per permettere ad **s1** di rimuovere tutti i diritti che il soggetto **s2** ha su un oggetto **f**.
2. Modificare il comando per permettere la rimozione solo se il soggetto **s1** ha o il permesso *own* oppure il permesso *modify* su **f**.
3. Modificare il comando per permettere la rimozione solo se **s1** ha il permesso *modify* su **f**, e **s2** non ha il permesso *own* su **f**

PROBLEMA 3 [2 x 5 punti]
Ci sono 3 proprietà desiderabili di una funzione hash crittografica:
a) resistente alla pre-immagine [data y, trovare una x tale che f(x)=y] ,

b) resistente ad altra pre-immagine [dato y=f(x) per qualche x, trovare una x' tale che f(x')=y]
c) resistente alle collisioni [trovare x e x' diversi tali che f(x)=f(x')].
Per ciascuna delle applicazioni seguenti, indicare una proprietà necessaria ed una superflua, **spiegando la scelta fatta**

1. Le password sono conservate in un file delle password in formato hash. Per autenticare uno studente, la password presentata da un utente è trasformata (con funzione hash) e confrontata con il valore hash memorizzato. Un utente che mette le mani sul file non dovrebbe essere in grado di autenticarsi in questo modo.
2. Un amministratore di sistema, preoccupato di possibili intromissioni, calcola il valore hash di file binari importanti e conserva i valori hash in un file read-only. Un programma ricalcola periodicamente i valori hash dei file che contengono i binari e li confronta con i valori conservati. Un utente malizioso che riuscisse a sovrascrivere uno dei file protetti non dovrebbe essere in grado di cambiarli senza essere scoperto.

PROBLEMA 4 [10 punti]
Per evitare di essere scoperto da un antivirus, il corpo di un virus è 'offuscato' sostituendolo con il risultato di un XOR con una sequenza T di byte costruita ripetendo varie volte una chiave segreta K di 8 byte (diversa per diverse istanze del virus). Per semplicità la lunghezza del codice del virus è un multiplo di 8. Il virus infetta un programma copiandosi in una locazione non prevedibile. Un file infetto contiene un "loader" che, se invocato durante l'esecuzione del file infetto, legge K, costruisce T, lo usa per estrarre il virus e ne 'lancia' il codice. Il codice del loader, la chiave K ed il codice del virus sono copiate in posizioni casuali nei file che vengono infettati. Il loader è un frammento di codice che può essere presente normalmente anche in file non infetti, e quindi NON è caratteristico del virus.
Siete riusciti ad ottenere una copia non offuscata del codice del virus e volete scoprire se vi sono occorrenze di questo virus in un insieme di file che si sospetta siano infetti. Come affrontereste questo problema?

PROBLEMA 5 [3 x 5 punti]
Un blocco di plaintext di 380 bit è criptato con DES con una delle modalità di cifratura producendo un blocco di ciphertext di 384 bit. Il ciphertext viene spedito al ricevente per essere decifrato. Durante la trasmissione, il bit 193 viene cambiato.

a) quanti bit POTREBBERO essere sbagliati dopo la decifratura usando CBC?
b) quanti bit POTREBBERO essere sbagliati dopo la decifratura usando CFB nella versione con blocchi di 32 bit (cioè 32 bit sono criptati ad ogni turno)?
c) quanti bit POTREBBERO essere sbagliati dopo la decifratura usando OFB nella versione con blocchi di 64 bit (cioè 64 bit sono criptati ad ogni turno)?

PROBLEMA 6 [10 punti]
Un programmatore vuole utilizzare CBC per proteggere sia l'integrità che la confidenzialità della trasmissione. Aggiunge un blocco P_{n+1} di bit zero ridondanti alla fine del plaintext e poi cifra con CBC. Ricevendo i blocchi cifrati, verifica che il blocco di bit ridondanti dopo la decifratura siano ancora zero. E' sufficiente questa verifica per confermare l'integrità del messaggio trasmesso?

PROBLEMA 7 [10 punti]
Definire una procedura per trasformare una Biba Strict Integrity Policy (senza la regola riguardante l'esecuzione) in una policy basata su RBAC. Argomentare la correttezza della trasformazione.