Cognome:_____ Nome: _____ Matricola:_____

Cybersecurity – 22 settembre 2020 – 4 CFU (la tesina vale 2 CFU)

SE NON HAI CONSEGNATO LA TESINA NON PUOI SOSTENERE QUESTO ESAME

Usa questa pagina per la brutta, staccala, non consegnarla.

Cognome:_____ Nome: _____ Matricola:_____

Cybersecurity – 22 settembre 2020 – 4 CFU (la tesina vale 2 CFU)

Usa questa pagina per la brutta, staccala, non consegnarla.

Cognome:_____ Nome: _____ Matricola:_____

Cybersecurity – 22 settembre 2020 – 4 CFU (la tesina vale 2 CFU)

SE NON HAI CONSEGNATO LA TESINA NON PUOI SOSTENERE QUESTO ESAME

Tempo a disposizione: 60 minuti. Libri e appunti chiusi. Vietato comunicare con chiunque. Vietato l'uso di smartphone/watch, calcolatrici e affini.

1. Discuti brevemente i principi di progettazione "eterogeneità" e "usabilità". Tra loro c'è una sinergia o un antagonismo? Spiega.

eterogeneità

usabilità

sinergia o antagonismo? Spiega.

2. Chiavi simmetriche

2.1. Qual è il motivo per cui è consigliabile cambiare una chiave simmetrica anche se non è stata divulgata?

2.2. In cosa consiste il problema della distribuzione delle chiavi simmetriche? Come si risolve, in genere?

2.3. Se volessimo generare una nuova chiave simmetrica casualmente, che precauzioni dovremmo prendere nella scelta del generatore di numeri pseudo-casuali e nel suo utilizzo?

2.4. Che cosa significa key-rollover? fornisci un esempio di come si possa fare key rollover.

key-rollover.

esempio

Cognome:_____ Nome: _____ Matricola:_____

Cybersecurity – 22 settembre 2020 – 4 CFU (la tesina vale 2 CFU)

3. Strutture dati autenticate.

3.1. Descrivi la struttura dati autenticata chiamata Merkle Hash Tree (MHT), aiutati con un **disegno** della struttura. Contestualmente descrivi come è fatta la **prova di integrità** del risultato di una query su un MHT rispetto ad un **root-hash fidato** e l'algoritmo di veririca.

disegno MHT	prova di integrità e algoritmo di verifica

3.2. Supponi che un client voglia **aggiornare** una ADS tenuta da un server non fidato. Il client ha il root-hash. Descrivi le operazioni lato client e le interazioni col server che portano all'aggiornamento dell'ADS e del root-hash tenuto dal client.

4. Sicurezza del codice.

Considera il seguente codice C eseguito con input non fidato in ambiente Unix.

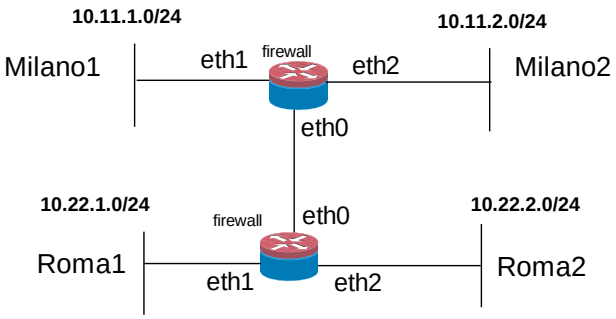
```
int main(int argc, char** argv) {
  char cmd[1000];
  char* outfile = getenv("OUTFILE");/*ottiene il contenuto della variabile di ambiente $OUTFILE*/
  strcpy(cmd, "report ");/* "report <arg>" è un comando fidato che genera output nel file <arg>*/
  strcat(cmd, outfile);
  system(cmd);
}
```

4.1. Elenca le vulnerabilità che pensi siano presenti in questo codice con una breve descrizione del problema.

4.2. Suggerisci delle modifiche al codice per risolvere le vulnerabilità.

5. Networking.

Considera la rete in figura. Non vi sono nat ed il routing è configurato correttamente.



5.1. Considera la policy in tabella in cui Q rappresenta la possibilità di iniziare connessioni (query) e R la possibilità di rispondere (Reply).

A / Da	Roma1	Roma2	Milano1	Milano2
Roma1	-----	Q	R	
Roma2	R	-----		Q
Milano1	Q		-----	
Milano 2		R		-----

Scrivi le configurazioni dei due firewall per realizzare la policy in tabella usando preferibilmente la sintassi di netfilter.

Roma

Milano

Cognome:_____ Nome: _____ Matricola:_____

Cybersecurity – 22 settembre 2020 – 4 CFU (la tesina vale 2 CFU)

6. Gli attacchi al login possono essere classificati in on-line (in cui il prompt di login è accessibile via rete) e off-line (in cui il database utenti/password è disponibile in locale all'attaccante con gli hash delle password). Compila la seguente tabella.

	On-line	Off-line
Che contromisure suggerisci per i due tipi di attacchi?		
Quali sono secondo te le criticità o difficoltà principali per l'attaccante?		
Quali sono secondo te le criticità o difficoltà principali per chi deve mitigare il rischio di un tale tipo di attacchi?		