

**Cognome:** \_\_\_\_\_ **Nome:** \_\_\_\_\_ **Matricola:** \_\_\_\_\_

**Sicurezza dei sistemi informatici e delle reti – 19 luglio 2007**

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

**Cognome:** \_\_\_\_\_ **Nome:** \_\_\_\_\_ **Matricola:** \_\_\_\_\_

**Sicurezza dei sistemi informatici e delle reti – 19 luglio 2007**

Cognome: \_\_\_\_\_ Nome: \_\_\_\_\_ Matricola: \_\_\_\_\_

**Sicurezza dei sistemi informatici e delle reti – 19 luglio 2007**

Tempo a disposizione: **90 minuti**. Libri e appunti chiusi. Vietato comunicare con chiunque. Vietato l'uso di cellulari, calcolatrici, palmari e affini.

1. Considera il seguente codice C relativo all'eseguibile "pippo" e rispondi alle seguenti domande.

```
int main(int argc, char** argv)
{
  char b[100];
  char a[1000];
  char* c;
  char d[101];
  scanf("%9999s", a);
  if (argc >= 2)
    strncpy(b, argv[1], 1000); /*copia da argv[1] in b*/
  c=getenv("INPUT");
  strncpy(d, c, 100); /* copia da c in d */
  ...
}
```

1.1. Sottolinea il codice che secondo te dà luogo a vulnerabilità e descrivi schematicamente il problema.

scanf("%9999s", a); legge da standard input 9999 caratteri in un buffer di 1000  
strncpy(b, argv[1], 1000); copia dal primo parametro della command line 1000 caratteri in un buffer di 100.

1.2. Supponi che un programma pluto richiami pippo, ad esempio usando la system call `execve()`. Pluto ottiene tre stringhe X, Y, Z in input dall'utente (fonte non fidata) che passa a pippo con le seguenti modalità

- i primi 999 caratteri di X sono passati a pippo tramite standard input, seguiti da un "\n" (a capo)
- i primi 999 caratteri di Y sono passati a pippo come primo parametro (`argv[1]`)
- i primi 100000 caratteri di Z sono passati a pippo come contenuto della variabile di ambiente "INPUT".

Un utente malevolo che usi il programma pluto che vulnerabilità di pippo riesce a sfruttare?  
Perché

X: solo i primi 999 caratteri sono passati a pippo che non controlla la lunghezza dell'input ma l'input entra sempre in a (1000 caratteri).

Y: vengono passati a pippo 999 caratteri che li copia in b che e' lungo 100

Z: vengono passati 1000000 caratteri ma pippo ne copia solo 100.

Si riesce a sfruttare solo la vulnerabilita' relativa a `strncpy(b, argv[1], 1000)`;

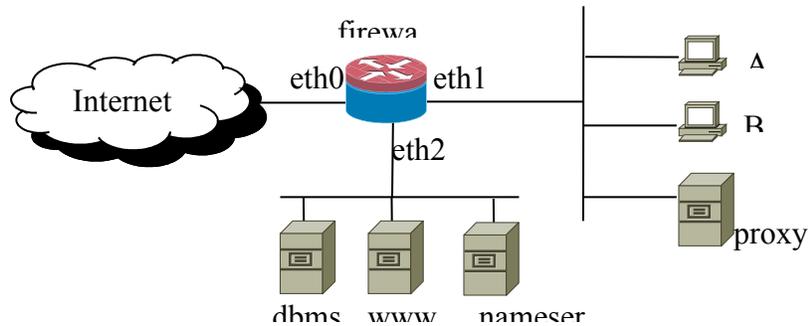
1.3. Supponi che ti venga richiesto di mettere in sicurezza il sistema ma non hai possibilità di modificare gli eseguibili né di pluto né di pippo che approccio puoi usare? Descrivilo.

Si puo' usare l'approccio del wrapper che verifica ciascun input prima di passarlo a pippo. In questo modo non c'e' bisogno di modificare ne pippo ne pluto. Si deve pero' far in modo che il wrapper appaia a pluto come se fosse pippo. Questo e' normalmente possibile con operazioni solo a livello di sistema operativo.

Cognome: \_\_\_\_\_ Nome: \_\_\_\_\_ Matricola: \_\_\_\_\_

**Sicurezza dei sistemi informatici e delle reti – 19 luglio 2007**

2. Considera la rete in figura con la matrice di accesso indicata, in cui i servizi di interesse sono tre: dns, web, e db.



	A	A	B	Proxy	Dbms	www	Nameserver	Internet
Da								
A			-	Rich. web	Rich. db	-	Rich. dns	-
B				Rich web	Rich. db	-	Rich. dns	-
Proxy	Rich. web	Rich. web			-	Rich. web	Rich. dns	Rich. web
Dbms	Rich. db	Rich. db				Rich. db	-	-
www				Rich. web	Rich. db		-	Rich. web
Nameserver	Rich. dns	Rich. dns		Rich. dns				Rich./Rich. dns
Internet				Rich. web		Rich. web	Rich./Rich. dns	

Rispondi alle seguenti domande.

2.1. Evidenzia sulla matrice di accesso le parti che non sono realizzabili per mezzo del firewall.

2.2. Che tecnica suggeriresti per realizzare le parti della matrice di accesso identificate nella domanda precedente? Cerca di dare una soluzione che non preveda l'acquisto di ulteriori apparati.

Dei personal firewall sulle macchine che ospitano i vari servizi.

2.3. Per l'implementazione pensi che possa essere impiegato facilmente un firewall stateless? Perché?

No, poiché la matrice di accesso da' le specifiche a livello di "sessioni" tcp e udp. Un firewall stateless sfrutta dei trucchi per capire lo stato delle sessioni dai pacchetti stessi (es. syn bit) ma tali tecniche non sono affidabili.

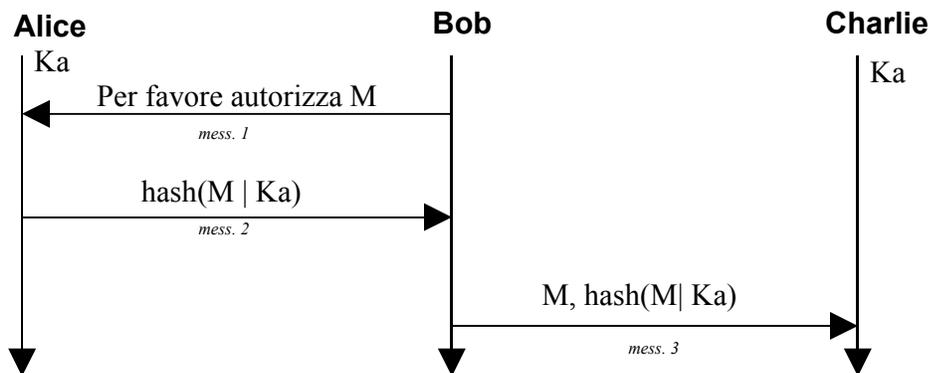
Cognome: \_\_\_\_\_ Nome: \_\_\_\_\_ Matricola: \_\_\_\_\_

Sicurezza dei sistemi informatici e delle reti – 19 luglio 2007

2.4. Dai la configurazione del firewall, usando preferibilmente la sintassi di netfilter delle parti sottolineate della matrice di accesso. Al posto degli indirizzi usa i nomi delle macchine, al posto delle porte usa i nomi dei servizi (dbms, dns, www).

```
:FORWARD DROP  
-A FORWARD -i eth1 -o eth0 -p tcp -s A -d dbms --dport db -m state --state NEW -j ACCEPT  
-A FORWARD -i eth0 -o eth0 -p udp -d nameserver --dport dns -m state --state NEW -j ACCEPT  
-A FORWARD -m state --state ESTABLISHED -j ACCEPT  
COMMIT
```

3. Bob, per chiedere a Charlie di effettuare una certa azione, deve dimostrare che Alice e' d'accordo. La richiesta è contenuta in un messaggio M in linguaggio naturale creato da Bob. Il protocollo usato è il seguente, dove Ka e' un segreto condiviso tra Alice e Charlie.



Se l'hash inviato in messaggio 3 e' corretto Charlie accetta la richiesta. Rispondi alle seguenti domande

3.1. Supponi che Alice voglia far credere a Charlie che Bob abbia richiesto di autorizzare il messaggio M'. Pensi sia possibile? Come?

**Si, Alice può impersonare Bob poiché la comunicazione non è autenticata.**

3.2. Suggestisci un protocollo, sempre basato su chiave simmetrica, che non abbia il problema riscontrato al punto precedente.

**Si può modificare il protocollo nel seguente modo. Bob e Charlie condividono una chiave segreta Kb. Il messaggio 3 diviene: M, hash(M|Kb), hash(M|Ka).**

Cognome: \_\_\_\_\_ Nome: \_\_\_\_\_ Matricola: \_\_\_\_\_

Sicurezza dei sistemi informatici e delle reti – 19 luglio 2007

3.3. Supponi che ci siano un gran numero di richieste che Alice autorizza facilmente (messaggi buoni) e un gran numero di richieste che Alice non autorizza (messaggi cattivi). Bob vuol far credere a Charlie che Alice ha autorizzato un messaggio cattivo qualsiasi. La lunghezza dell'hash è 64 bit ma per Bob è possibile fare in tempi ragionevoli una ricerca su uno spazio di ricerca di  $2^{34}$  elementi. Pensi che Bob possa riuscire nell'impresa? In caso affermativo che approccio potrebbe adottare?

Si potrebbe pensare che Bob possa tentare un attacco a compleanno. Infatti il testo è libero e quindi Bob ha ampia libertà di sceglierlo (vedi dispense per un esempio). Lo scopo di Bob sarebbe quello di trovare due messaggi uno cattivo e uno buono con lo stesso hash. Vi è alta probabilità di trovare tali messaggi in uno spazio grande  $1.2 * 2^{32}$  e Bob può fare ricerche in uno spazio  $2^{34}$ .

Tuttavia, in questo caso, l'attacco non si può fare poiché l'hash non viene fatto sul messaggio ma sul messaggio concatenato ad una chiave che non si conosce.

Se l'autenticazione fosse effettuata mediante firma digitale con chiave asimmetrica l'attacco a compleanno sarebbe possibile, in quel caso infatti l'hash firmato viene calcolato sul solo messaggio.

4. Rispondi alle seguenti domande circa la pianificazione della sicurezza.

4.1. Elenca almeno tre motivi per cui è utile pianificare la sicurezza

Vedi materiale didattico

4.2. Elenca in sintesi le parti che ritieni fondamentali in un piano di sicurezza.

Vedi materiale didattico

4.3. In che rapporto sono il Documento Programmatico di Sicurezza previsto dalla legge 196/2003 e il piano di sicurezza?

	Piano disicurezza	DPS
obiettivi	evitare danni economici all' organizzazione	documentare le misure di sicurezza per i dati personali
obbligatorietà	a discrezione del management	obbligatorio per legge
struttura	libera, linee guida	alcune parti sono obbligatorie per legge
rapporto tra i contenuti	può prevedere la redazione del DPS tra i requisiti di conformità alla normativa	contiene parti tipiche di un piano di sicurezza, non prevede la redazione di un piano di

Cognome: \_\_\_\_\_ Nome: \_\_\_\_\_ Matricola: \_\_\_\_\_

Sicurezza dei sistemi informatici e delle reti – 19 luglio 2007

vigente (in Italia)

sicurezza generale

5. Considera i seguenti file e directory in un filesystem unix con i loro permessi. Gli utenti Pluto e Paperino appartengono al gruppo Disney, mentre Pippo non appartiene a tale gruppo.

Permessi	Utente	Gruppo	Path
d r-x r-x r-x	Root	Root	/
- rw- r-- ---	Pippo	Disney	/file1
d r-x rwx r-x	Paperino	Disney	/repository
- r-- rw- rw-	Paperino	Disney	/repository/file2
- --- --- r--	Pluto	Disney	/repository/file3

Supponendo i permessi immutabili, esprimi con una matrice di accesso su  $S=\{\text{Pippo, Pluto, Paperino}\}$ ,  $O=\{\text{file1, file2, file3, repository}\}$  lo stato del sistema. I diritti da indicare nella matrice sono: per i file  $R_{\text{file}}=\{\text{R (read), W (write), D (cancella il file), N (rinomina)}\}$ , per la directory  $R_{\text{dir}}=\{\text{L (lista della directory), C (creazione di un nuovo file nella directory)}\}$ .

	File1	File2	File3	repository
Pippo	RW	RW	R	L
Pluto	R	RWDN	DN	LC
Paperino	R	R		L

6. Cosa è SELinux? in che senso è considerato “più sicuro” di Linux?

SELinux è una opzione di compilazione di Linux che attiva nel kernel un MAC aggiuntivo rispetto al classico DAC di unix. Le policy MAC sono configurabili in maniera molto flessibili e permettono un confinamento molto fine. Se le policy sono ben scritte un problema in un software non si può propagare all'intero sistema. In pratica la sicurezza di un sistema SELinux dipende dalle policy e da eventuali vulnerabilità del kernel e non da vulnerabilità dei software applicativi e delle librerie.

7. Se leggi su un data sheet che un prodotto è certificato Common Criteria “eal 4”, pensi che questa informazione sia sufficiente per valutare il suo livello di sicurezza? Quali ulteriori indagini suggeriresti? Illustra brevemente il concetto di “livello di garanzia” in Common Criteria rispetto a questo problema.

Non è sufficiente. Bisogna leggere il Security Target. CC valuta se il Target Of Evaluation (cioè il prodotto in una certa configurazione) è conforme agli obiettivi di sicurezza stabiliti nel Security Target. Il livello di garanzia (es. eal4) stabilisce con quanta meticolosità le verifiche sono state condotte.

Cognome: \_\_\_\_\_ Nome: \_\_\_\_\_ Matricola: \_\_\_\_\_

Sicurezza dei sistemi informatici e delle reti – 19 luglio 2007

8. Considera la seguente matrice di accesso.  $S=\{u1, u2, u3, u4, u5\}$ ,  $O=\{f1, f2, f3, f4, f5\}$ ,  $R=\{r, w, x, g\}$  (read, write, execution, grant **con attenuazione del privilegio**, cioè non si può cedere diritti che non si hanno).

	f1	f2	f3	f4
u1	r	r	g	
u2	w	w		
u3		r	wg	r
u4			r	w

Rispondi alle seguenti domande.

8.1. La politica mostrata è MAC, DAC o altro? motiva la risposta.

mac      **dac**      altro

Motivazione:

**grant permette all'utente u3 di modificare i permessi sull'oggetto f3 a favore di altri utenti.**

8.2. Pensi sia possibile per u1 comunicare informazioni a u4 **eventualmente con la collaborazione di altri utenti?** in che modo?

**Si. u3 dà i diritti di scrittura su f3. u1 scrive le informazioni su f3. u4 legge da f3.**

8.3. Una “comunità” è un insieme di soggetti tali che ciascuna coppia di soggetti di una comunità può scambiare informazioni (anche indirettamente) in entrambi i versi. Quali comunità nascono dalla matrice di accesso mostrata nel caso in cui nessuno applichi il diritto di grant?

**{u1}    {u2}    {u3,u4}**

8.4. Quali comunità nascono dalla matrice di accesso mostrata nel caso in cui si possa applicare il diritto di grant?

**le stesse**

**{u1}    {u2}    {u3,u4}**

8.5. Se i diritti di u3 su f3 fossero “rwg” quali comunità nascerebbero dall’applicazione di grant?

**Una sola contenente tutti {u1, u2, u3, u4} infatti u3 può dare diritto di scrittura e lettura a tutti su f3 e quindi chiunque può comunicare con chiunque tramite f3.**