Université de BOUMERDES UMBB

Sécurité informatique
Responsable du Module : Mr RIAHLA Med Amine

TD N° 3 : Menaces (failles de sécurité, Attaques et vulnérabilités)

Exercice 1 « Messagerie »

En ouvrant une session Telnet sur le port 25 de son serveur SMTP (Simple Mail Transfer Protocol), il est possible d'envoyer un courrier électronique avec un expéditeur fantaisiste.

Illustrer cette technique en utilisant les commandes SMTP : HELO, MAIL FROM :, RCPT TO :, DATA pour envoyer un courrier forgé.

Exercice 2 « Virus et Malwares »

1. Quelle est la différence entre un virus et un vers ?
2. Dans quelle mesure les vers sont ils plus dangereux que les virus ?
3. Certains vers qui se propagent sur internet ne provoquent aucun dommage sur les machines atteintes. Pourquoi sont-ils cependant nuisibles ?
4. Pour désinfecter un ordinateur, il est recommandé de le redémarrer depuis une disquette. Pourquoi ?
5. Comment un pirate peut il procéder pour installer une porte dérobé ?
6. Comment un pirate peut il procéder pour installer un cheval de Troie ?
7. Décrire deux techniques différentes qui permettent à des virus de ne pas être détecter par des logiciels antivirus

Exercice 3 « Virus avec fichier joint chiffré»

On considère dans cet exercice une variante du vers W32/Beagle. Ce ver se présente sous forme d'un courrier électronique possédant un fichier joint, qui est à la fois compressé et chiffré. Le mot de passe pour déchiffrer le fichier est contenu dans le corps du message. Si la victime exécute le fichier obtenu après décompression avec le mot de passe fourni (qui est un fichier avec une extension .exe). Alors le ver se propage en choisissant la prochaine victime dans le carnet d'adresse de la victime courant.

Pourquoi le fichier compressé est il chiffré puisque le mot de passe est fourni dans le message ?

Exercice 4 « Les pointeurs en C »

Quelles valeurs le programme en C suivant va-t-il afficher ? Pourquoi ?

```
#include<stdio.h>

void main() {
        char buffer[10];
        char *ptr;

        buffer[0] = 'A';
        buffer[1] = 'B';
        buffer[2] = 'C';
        buffer[3] = 'D';
        ptr = buffer + 2;
        *ptr = 'Z';
        printf("%c %c %c %c \n", buffer[0], buffer[1], buffer[2], buffer[3]);
}
```

Exercice 5 « Modification d'adresse dans la pile »

1. Quelle valeur le programme en C ci-dessous va-t-il afficher ?pourquoi ?
2. Dessiner un diagramme de la pile en considérant que toutes les variables sont alignées sur des multiples de 4 octets et que les adresses sont stockées sur 4 octets.
3. A quoi correspondent les valeurs 12 et 10 dans la procédure « function »?

```
#include<stdio.h>
void function(int a, int b, int c)
{
        char buffer1[5];
        char buffer2[10];
        char *ptr;

        ptr = buffer1 + 12;
        *ptr += 10;
}
void main()
{
        int x;

        x = 0;
        function(1,2,3);
        x = 1;
        printf("%d\n",x);
}
```

Exercice 6 « Exploit d'un programme en C »

La société Dypofloo est spécialisée dans la vente de photos numériques sur internet. Les personnes souhaitant acheter des photos sur le site web de l'entreprise doivent avoir préalablement ouvert un compte au prêt de la société. Lorsque un client souhaite avoir accès à une photo qu'il souhaite acheter, il exécute via son navigateur un script qui appelle la fonction acheter décrite ci-dessous. Cette fonction prend en argument l'identifiant du client (login), son mot de passe (password), son nom (nom) ainsi que le numéro de photo désiré (numero). La fonction débiter comptabilise les photos auxquelles le client a accéder, la fonction afficher retourne la photo sur le navigateur du client et la fonction authentifier permet de vérifier que le mot de passe du client est correct. On suppose que ces trois fonctions, qui ne sont pas données ici, ont été correctement implémentées.

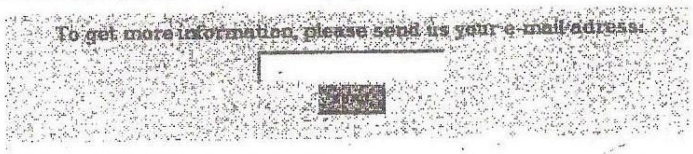
```
void buy(const char* login, const char* password, const char* name, const char* number)
{
    if (authenticate(login, password)==1)
    {
        message display picture(name, number);
        message debit(login);
    }
}
void message display picture(const char* name, const char* number)
{
    char message[100]="";
    strcat (message, "Dear ");
    strcat (message, name);
    strcat (message, ", here is the picture you requested.\n");
    printf (message);
    display(number);
}
void message debit(const char* login)
{
    debit(login);
    printf("10 Euros have been charged from your account.\n");
}
```

1. Quelle technique est fréquemment employée pour abuser d'un programme en particulier en C ?
2. Décrire comment cette technique peut être appliquée dans le cas présent afin qu'un client puisse accéder aux photos sans être débité du montant de l'achat.

Exercice 7 « Vulnérabilités des scripts CGI »

Lorsque l'on navigue sur le Web, on est amené à remplir des formulaires en ligne. Les données de ces formulaires sont alors envoyées au serveur et traitées par un programme CGI (Common Gateway Interface). Les langages les plus utilisées actuellement pour écrire des programmes CGI sont Perl, Php, C, ASP et encor le langage shell

Voici la copie partielle d'écran d'une page web



Voici le code html de cette page web

```
<HTML>
<BODY bgcolor="#FAF0E6">
<BR><BR><BR><BR>
<P align="center">
<HR noshade>
<TABLE align="center">
<TR><TD align="center">
To get more information, please send us your e-mail address:
</TD></TR>
</TD></TR>
<TR><TD align="center">
<FORM ACTION="/cgi-bin/mail.pl" METHOD=POST>
<INPUT TYPE="text" NAME="mail"><BR>
<INPUT TYPE=SUBMIT VALUE="send">
</FORM>
</TD></TR>
</TABLE>
<HR noshade>
</P>
</BODY>
</HTML>
```

Voici enfin le programme CGI écrit en Perl qui trait les données

```
#!/usr/bin/perl
use CGI;
my $q = new CGI;
my $address = $q->param ("mail");
open MAIL, "| /usr/lib/sendmail $address";
print MAIL "To: $address \n";
print MAIL "From: ATM and Co\n\n";
print MAIL "We have received your request, thank you very much.\n";
print MAIL "You will receive our documentation by mail shortly.\n";
close(MAIL);
print "Content-type: text/html\n\n";
print "<HTML>";
print "<BODY bgcolor=\"#FAF0E6\">";
print "<P align=\"center\"><A href=\"/index.html\">Back to the summary</A></P>";
print "</BODY>";
print "</HTML>";
```

1. Quel est l'objectif du programme CGI tel qu'il a été prévu par le concepteur du site web ?
2. Les possibilités d'action d'un pirate sont restreintes puisqu'il ne peut que remplir le champ du formulaire. Intuitivement, que peut il tenter ?
3. Comment peut il se faire envoyer par courrier électronique le fichier des mots de passe (/etc/passwd) du server HTTP.

Exercice 8 « Serveur WEB»

Les logs d'un serveur web contiennent une série d'entrées du type [adresse source, date, requête, résultat, octets transférés], expliquer la particularité du log suivant :

```
128.178.146.216 - - [24/Sep/2003:16:50:42 +0200] "GET
/default.ida?XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX%u9090%u6858%ucbd3%u7801%u9
090%u6858%ucbd3%u7801%u9090%u6858%ucbd3%u7801%u9090%u9090%u8190%u00c3%u
0003%u8b00%u531b%u53ff%u0078%u0000%u00=a %HTTP/1.0" 404 209
```