

# [UNIX] Samba Server Multiple Vulnerabilities

---

*Source:* <http://www.derkeiler.com/Mailing-Lists/Securiteam/2007-02/msg00021.html>

---

- *From:* SecuriTeam <[support@xxxxxxxxxxxxxx](mailto:support@xxxxxxxxxxxxxx)>
  - *Date:* 7 Feb 2007 17:47:53 +0200
- 

The following security advisory is sent to the securiteam mailing list, and can be found at the SecuriTeam web site: <http://www.securiteam.com>  
-- promotion

The SecuriTeam alerts list – Free, Accurate, Independent.

Get your security news from a reliable source.  
<http://www.securiteam.com/maillinglist.html>

-----

Samba Server Multiple Vulnerabilities

---

## SUMMARY

" <<http://www.samba.org/>> Samba is a free software re-implementation of SMB/CIFS networking protocol released under the GNU General Public License. As of version 3, Samba not only provides file and print services for various Microsoft Windows clients but can also integrate with a Windows Server domain, either as a Primary Domain Controller (PDC) or as a Domain Member. It can also be part of an Active Directory domain."

Three vulnerabilities were discovered in Samba Server: Format string, infinite loop in smbd and potential buffer overrun.

## DETAILS

Format string bug in afsacl.so VFS plugin:

Versions:

The AFS ACL mapping VFS plugin distributed in Samba 3.0.6 – 3.0.23d (inclusive)

CVE Information:

<<http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2007-0454>>

## [UNIX] Samba Server Multiple Vulnerabilities

CVE-2007-0454

The name of a file on the server's share is used as the format string when setting an NT security descriptor through the afsacl.so VFS plugin.

This vulnerability only impacts Samba servers that share AFS file systems to CIFS clients and which have been explicitly instructed in smb.conf to load the afsacl.so VFS module.

The source defect results in the name of a file stored on disk being used as the format string in a call to sprintf(). This bug becomes exploitable only when a user is able to write to a share which utilizes Samba's afsacl.so library for setting Windows NT access control lists on files residing on an AFS file system.

### Patch:

A patch against Samba 3.0.23d has been attached to this email. This fix has been incorporated into the Samba 3.0.24 release.

Patches are also available from at the Samba Security page (<http://www.samba.org/samba/security>)

### Workaround:

An unpatched server may be protected by removing all references to the afsacl.so VFS module from shares in smb.conf.

### Disclosure Timeline:

- \* Jan 8, 2007: Defect first reported to the security@xxxxxxxxxx email alias.
- \* Jan 8, 2007: Initial developer response by Jeremy Allison confirming the issue.
- \* Jan 29, 2007: Announcement to vendor-sec mailing list
- \* Feb 5, 2007: Public issue of security advisory.

### afsacl.patch:

```
diff -urN samba-3.0.23d/source/modules/vfs_afsacl.c
samba/source/modules/vfs_afsacl.c
--- samba-3.0.23d/source/modules/vfs_afsacl.c 2006-06-23
08:16:50.000000000 -0500
+++ samba/source/modules/vfs_afsacl.c 2007-01-29 20:11:07.000000000
-0600
@@ -901,7 +901,7 @@
ZERO_STRUCT(dir_acl);
ZERO_STRUCT(file_acl);

- pstr_sprintf(name, fsp->fsp_name);
+ pstrcpy(name, fsp->fsp_name);

if (!fsp->is_directory) {
/* We need to get the name of the directory containing the
```

## [UNIX] Samba Server Multiple Vulnerabilities

Potential Denial of Service bug in smbd

Versions:

\* Samba 3.0.6 – 3.0.23d (inclusive)

CVE Information:

<<http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2007-0452>>  
CVE-2007-0452

Internally Samba's file server daemon, smbd, implements support for deferred file open calls in an attempt to serve client requests that would otherwise fail due to a share mode violation. When renaming a file under certain circumstances it is possible that the request is never removed from the deferred open queue. smbd will then become stuck in a loop trying to service the open request.

This bug may allow an authenticated user to exhaust resources such as memory and CPU on the server by opening multiple CIFS sessions, each of which will normally spawn a new smbd process, and sending each connection into an infinite loop.

Patch:

A patch against Samba 3.0.23d has been attached to this email. This fix has been incorporated into the Samba 3.0.24 release. Patches are also available from at the Samba Security page (<http://www.samba.org/samba/security>)  
<http://www.samba.org/samba/security>).

Workaround:

The bug is believed to be exploitable only by an authenticated user. The server's exposure can be alleviated by disabling any suspect or hostile user accounts.

smbd\_deferred\_open\_v2.patch:

```
diff -urN samba-3.0.23d/source/printing/nt_printing.c
samba/source/printing/nt_printing.c
--- samba-3.0.23d/source/printing/nt_printing.c 2006-07-10
11:27:50.000000000 -0500
+++ samba/source/printing/nt_printing.c 2007-01-30 15:00:45.000000000
-0600
@@ -4839,7 +4839,7 @@
pstrncpy( file, s );
driver_unix_convert(file, conn, NULL, &bad_path,
&st);
DEBUG(10,("deleting driverfile [%s]\n", s));
- unlink_internals(conn, 0, file, False);
+ unlink_internals(conn, 0, file, False, False);
}
}
```

## [UNIX] Samba Server Multiple Vulnerabilities

@@ -4848,7 +4848,7 @@

```
pstrcpy( file, s );
driver_unix_convert(file, conn, NULL, &bad_path,
&st);
DEBUG(10,("deleting configfile [%s]\n", s));
- unlink_internals(conn, 0, file, False);
+ unlink_internals(conn, 0, file, False, False);
}
}
```

@@ -4857,7 +4857,7 @@

```
pstrcpy( file, s );
driver_unix_convert(file, conn, NULL, &bad_path,
&st);
DEBUG(10,("deleting datafile [%s]\n", s));
- unlink_internals(conn, 0, file, False);
+ unlink_internals(conn, 0, file, False, False);
}
}
```

@@ -4866,7 +4866,7 @@

```
pstrcpy( file, s );
driver_unix_convert(file, conn, NULL, &bad_path,
&st);
DEBUG(10,("deleting helpfile [%s]\n", s));
- unlink_internals(conn, 0, file, False);
+ unlink_internals(conn, 0, file, False, False);
}
}
```

@@ -4882,7 +4882,7 @@

```
pstrcpy( file, p );
driver_unix_convert(file, conn, NULL,
&bad_path, &st);
DEBUG(10,("deleting dependent file
[%s]\n", file));
- unlink_internals(conn, 0, file, False);
+ unlink_internals(conn, 0, file, False,
False);
}
```

i++;

```
diff -urN samba-3.0.23d/source/smbd/nttrans.c samba/source/smbd/nttrans.c
--- samba-3.0.23d/source/smbd/nttrans.c 2006-06-23 08:16:49.000000000
-0500
```

```
+++ samba/source/smbd/nttrans.c 2007-01-30 15:00:45.000000000 -0600
```

@@ -664,7 +664,7 @@

```
if (lp_acl_check_permissions(SNUM(conn)) && (share_access &
FILE_SHARE_DELETE)
&& (access_mask & DELETE_ACCESS)) {
#endif
```

## [UNIX] Samba Server Multiple Vulnerabilities

```
- status = can_delete(conn, fname, file_attributes,
bad_path, True);
+ status = can_delete(conn, fname, file_attributes,
bad_path, True, False);
/* We're only going to fail here if it's access denied, as
that's the
only error we care about for "can we delete this ?"
questions. */
if (!NT_STATUS_IS_OK(status) &&
(NT_STATUS_EQUAL(status,NT_STATUS_ACCESS_DENIED) ||
@@ -1281,7 +1281,7 @@
/* Setting FILE_SHARE_DELETE is the hint. */
if (lp_acl_check_permissions(SNUM(conn)) && (share_access &
FILE_SHARE_DELETE) && (access_mask & DELETE_ACCESS)) {
#endif
- status = can_delete(conn, fname, file_attributes,
bad_path, True);
+ status = can_delete(conn, fname, file_attributes,
bad_path, True, False);
/* We're only going to fail here if it's access denied, as
that's the
only error we care about for "can we delete this ?"
questions. */
if (!NT_STATUS_IS_OK(status) &&
(NT_STATUS_EQUAL(status,NT_STATUS_ACCESS_DENIED) ||
@@ -1888,8 +1888,14 @@

status = rename_internals(conn, fsp->fsp_name,
new_name, 0, replace_if_exists,
path_contains_wcard);
- if (!NT_STATUS_IS_OK(status))
+
+ if (!NT_STATUS_IS_OK(status)) {
+ if (open_was_deferred(SVAL(inbuf,smb_mid))) {
+ /* We have re-scheduled this call. */
+ return -1;
+ }
return ERROR_NT(status);
+ }

/*
* Rename was successful.
diff -urN samba-3.0.23d/source/smbd/reply.c samba/source/smbd/reply.c
--- samba-3.0.23d/source/smbd/reply.c 2006-06-23 08:16:49.000000000
-0500
+++ samba/source/smbd/reply.c 2007-01-30 15:00:45.000000000 -0600
@@ -1865,7 +1865,7 @@
Check if a user is allowed to delete a file.
*****/

-NTSTATUS can_delete(connection_struct *conn, char *fname, uint32 dirty,
```

## [UNIX] Samba Server Multiple Vulnerabilities

```
BOOL bad_path, BOOL check_is_at_open)
+NTSTATUS can_delete(connection_struct *conn, char *fname, uint32 dirtytype,
BOOL bad_path, BOOL check_is_at_open, BOOL can_defer)
{
SMB_STRUCT_STAT sbuf;
uint32 fattr;
@@ -1938,7 +1938,7 @@
FILE_OPEN,
0,
FILE_ATTRIBUTE_NORMAL,
- 0,
+ can_defer ? 0 :
INTERNAL_OPEN_ONLY,
NULL);

if (!fsp) {
@@ -1960,7 +1960,7 @@
code.

*****/

-NTSTATUS unlink_internals(connection_struct *conn, uint32 dirtytype, char
*name, BOOL has_wild)
+NTSTATUS unlink_internals(connection_struct *conn, uint32 dirtytype, char
*name, BOOL has_wild, BOOL can_defer)
{
pstring directory;
pstring mask;
@@ -2000,7 +2000,7 @@
if (!has_wild) {
pstrcat(directory, "/");
pstrcat(directory, mask);
- error = can_delete(conn, directory, dirtytype, bad_path, False);
+ error =
can_delete(conn, directory, dirtytype, bad_path, False, can_defer);
if (!NT_STATUS_IS_OK(error))
return error;

@@ -2058,7 +2058,7 @@
}

sfprintf(fname, sizeof(fname)-1,
"%s/%s", directory, dname);
- error =
can_delete(conn, fname, dirtytype, bad_path, False);
+ error =
can_delete(conn, fname, dirtytype, bad_path, False, False);
if (!NT_STATUS_IS_OK(error)) {
continue;
}
@@ -2104,7 +2104,7 @@
```

## [UNIX] Samba Server Multiple Vulnerabilities

```
DEBUG(3,("reply_unlink : %s\n",name));

- status = unlink_internals(conn, dirtype, name,
path_contains_wcard);
+ status = unlink_internals(conn, dirtype, name,
path_contains_wcard, True);
if (!NT_STATUS_IS_OK(status)) {
if (open_was_deferred(SVAL(inbuf,smb_mid))) {
/* We have re-scheduled this call. */
diff -urN samba-3.0.23d/source/smbd/trans2.c samba/source/smbd/trans2.c
--- samba-3.0.23d/source/smbd/trans2.c 2006-11-14 08:42:12.000000000
-0600
+++ samba/source/smbd/trans2.c 2007-01-30 15:00:35.000000000 -0600
@@ -4446,9 +4446,15 @@
fname, newname ));
status = rename_internals(conn, fname,
base_name, 0, overwrite, False);
}
+
if (!NT_STATUS_IS_OK(status)) {
+ if
(open_was_deferred(SVAL(inbuf,smb_mid))) {
+ /* We have re-scheduled this call.
*/
+ return -1;
+ }
return ERROR_NT(status);
}
+
process_pending_change_notify_queue((time_t)0);
SSVAL(params,0,0);
send_trans2_replies(outbuf, bufsize, params, 2,
*ppdata, 0);
```

Buffer overrun in NSS host lookup Winbind library on Solaris:

Versions:

\* Samba 3.0.21 – 3.0.23d (inclusive) running on Sun Solaris

CVE Information:

<<http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2007-0453>>  
CVE-2007-0453

This vulnerability only affects Sun Solaris systems running Samba's winbindd daemon and configured to make use of the nss\_winbind.so.1 library for gethostbyname() and getipnodebyname() name resolution queries.

For example,

```
## /etc/nsswitch.conf
```

...

## [UNIX] Samba Server Multiple Vulnerabilities

ipnodes: files winbind  
hosts: files winbind

The buffer overrun is caused by copying a string passed into the NSS interface into a static buffer prior to sending the request to the winbindd daemon.

A patch against Samba 3.0.23d has been attached to this email. This fix has been incorporated into the Samba 3.0.24 release.  
Patches are also available from at the Samba Security page (<http://www.samba.org/samba/security>)  
(<http://www.samba.org/samba/security>).

### Workaround:

An unpatched Solaris server may be protected by removing the 'winbind' entry from the hosts and ipnodes services in /etc/nsswitch.conf.

### Disclosure Timeline:

- \* Dec 15, 2006: Defect first reported to the security@xxxxxxxx email alias.
- \* Dec 21, 2006: Initial developer response by Andrew Tridgell confirming the issue.
- \* Jan 29, 2007: Announcement to vendor-sec mailing list
- \* Feb 5, 2007: Public issue of security advisory.

### winbind\_nss\_solaris.patch:

```
diff -urN samba-3.0.23d/source/nsswitch/winbind_nss_solaris.c
samba/source/nsswitch/winbind_nss_solaris.c
--- samba-3.0.23d/source/nsswitch/winbind_nss_solaris.c 2006-04-19
21:29:21.000000000 -0500
+++ samba/source/nsswitch/winbind_nss_solaris.c 2007-01-29
19:51:11.000000000 -0600
@@ -493,7 +493,8 @@
af = AF_INET6;
#endif

- strncpy(request.data.winsreq, argp->key.name,
strlen(argp->key.name)) ;
+ strncpy(request.data.winsreq, argp->key.name,
sizeof(request.data.winsreq) - 1);
+ request.data.winsreq[sizeof(request.data.winsreq) - 1] = '\0';

if( (ret = winbindd_request_response(WINBINDD_WINS_BYNAME,
&request, &response))
== NSS_STATUS_SUCCESS ) {
@@ -515,7 +516,8 @@
ZERO_STRUCT(response);
ZERO_STRUCT(request);

- strncpy(request.data.winsreq, argp->key.name,
strlen(argp->key.name));
```

## [UNIX] Samba Server Multiple Vulnerabilities

```
+ strncpy(request.data.winsreq, argp->key.name,
sizeof(request.data.winsreq) - 1);
+ request.data.winsreq[sizeof(request.data.winsreq) - 1] = '\0';

if( (ret = winbindd_request_response(WINBINDD_WINS_BYNAME,
&request, &response))
== NSS_STATUS_SUCCESS ) {
```

### ADDITIONAL INFORMATION

The information has been provided by <<mailto:jerry@xxxxxxxx>> "Gerald (Jerry) Carter".

=====

This bulletin is sent to members of the SecuriTeam mailing list.  
To unsubscribe from the list, send mail with an empty subject line and body to:  
[list-unsubscribe@xxxxxxxx](mailto:list-unsubscribe@xxxxxxxx)  
In order to subscribe to the mailing list, simply forward this email to: [list-subscribe@xxxxxxxx](mailto:list-subscribe@xxxxxxxx)

=====  
=====

### DISCLAIMER:

The information in this bulletin is provided "AS IS" without warranty of any kind.  
In no event shall we be liable for any damages whatsoever including direct, indirect, incidental, consequential, loss of business profits or special damages.