Winning hardest hacking competition with advanced exploits

HACKING NAS AT PWN2OWN

Introduction

- Pwn2Own Austin 2021
  - Going as Synacktiv team
  - 3 working on Western Digital NAS:
    - @OnlyTheDuck
    - @JohnCool__
    - Etienne (the handsome guy)

My Cloud Pro Series PR4100 from WD	$40,000 (USD)	4
3TB My Cloud Home - Personal Cloud Storage from WD	$40,000 (USD)	4

Introduction

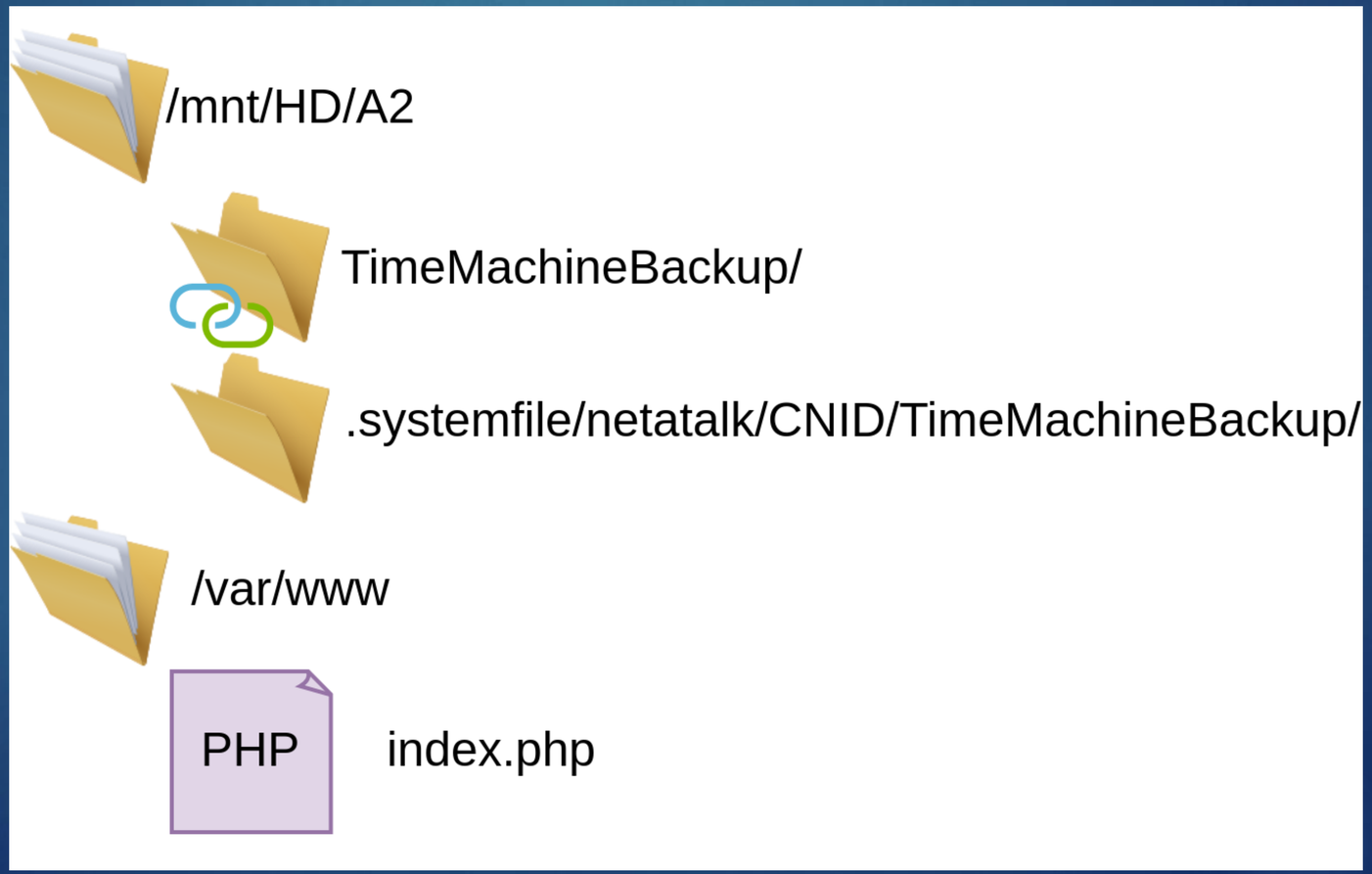
- Decided to target **netatalk**
- Linux implementation of the Apple File Protocol
  - Equivalent of SMB for Apple devices
  - Very old implementation
  - Used by **a LOT** of IoT devices
- By default accessible shared folders as guest
  - Seems juicy…

Function create_apple_desktop

```
olddtpath = bfromcstr(vol->v_path);
bcatcstr(olddtpath, "/" ".AppleDesktop");

dtpath = bfromcstr(vol->v_dbpath);
bcatcstr(dtpath, "/" ".AppleDesktop");

if (lstat(cfrombstr(dtpath), &st) != 0) {

    become_root();

    if (lstat(cfrombstr(olddtpath), &st) == 0) {
        cmd_argv[0] = "mv";
        cmd_argv[1] = bdata(olddtpath);
        cmd_argv[2] = bdata(dtpath);
        cmd_argv[3] = NULL;
        if (run_cmd("mv", cmd_argv) != 0) {
            LOG(log_error, logtype_afpd, "moving .AppleDesktop from \"%s\" to \"%s\" failed",
                bdata(olddtpath), bdata(dtpath));
            mkdir(cfrombstr(dtpath), 0777);
        }
    } else {
        mkdir(cfrombstr(dtpath), 0777);
    }

    unbecome_root();
}
```

Function create_apple_desktop

- **Function tries to mv the .AppleDesktop “directory” from shared folder to DB**



Function create_apple_desktop

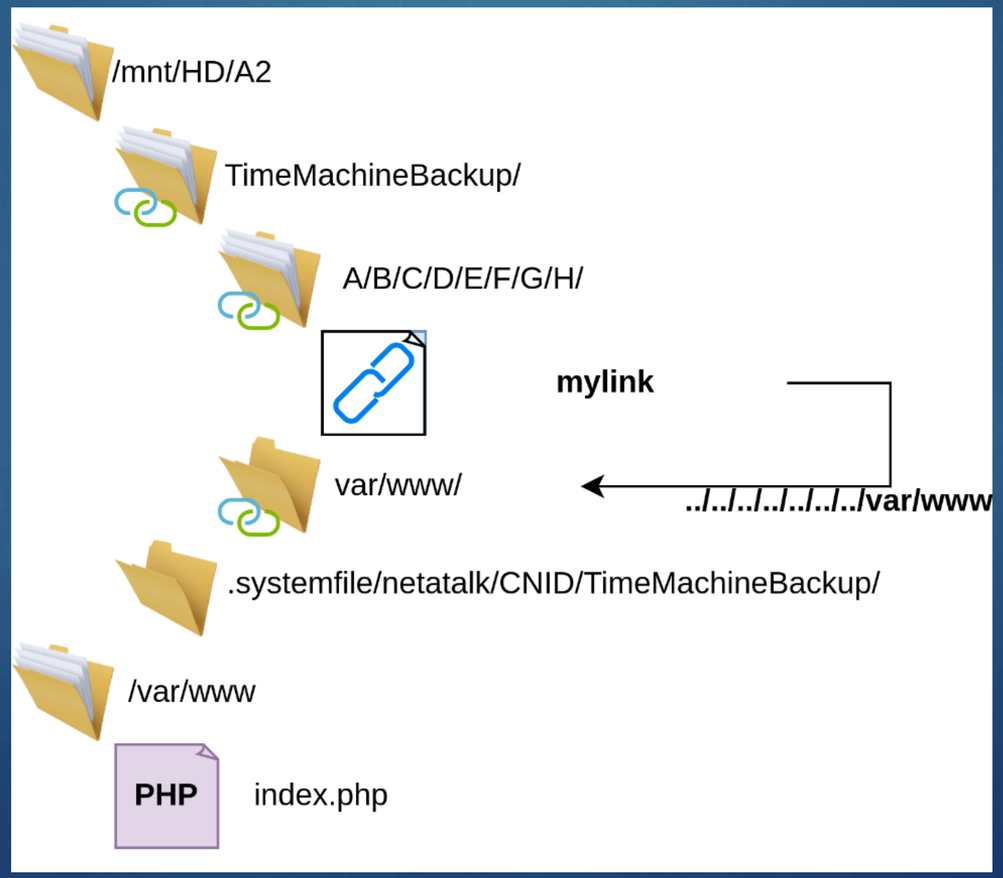
- **But cannot create files named .AppleDesktop using netatalk…**
- **But we can with SMB !**
  - Same shared folder used by both protocols !
  - Also accessible as guest, still no authentication

Capabilities

- AFP allows to:
  - Write files and directories inside the shared folder.
  - **Create symbolic links in the shared folder to arbitrary destinations.**
- SMB allows to:
  - Write files and directories inside the shared folder.
  - **Rename files, directories or even symbolic links.**

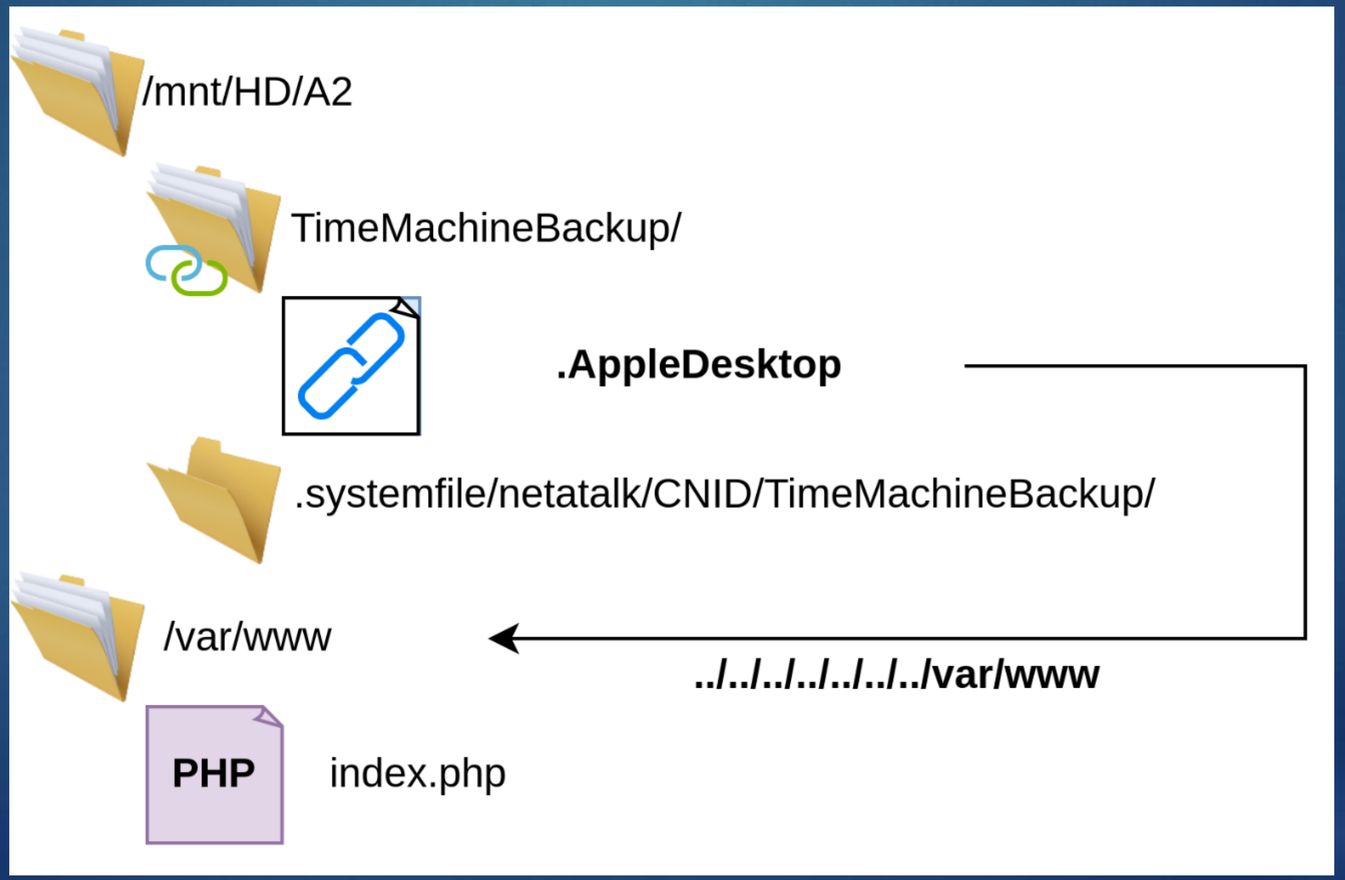
Exploit strategy

- Create a symlink A/B/C/D/E/F/G/mylink pointing to ../../../../../../../var/www
  - Points to a valid directory in the shared folder (var/www)



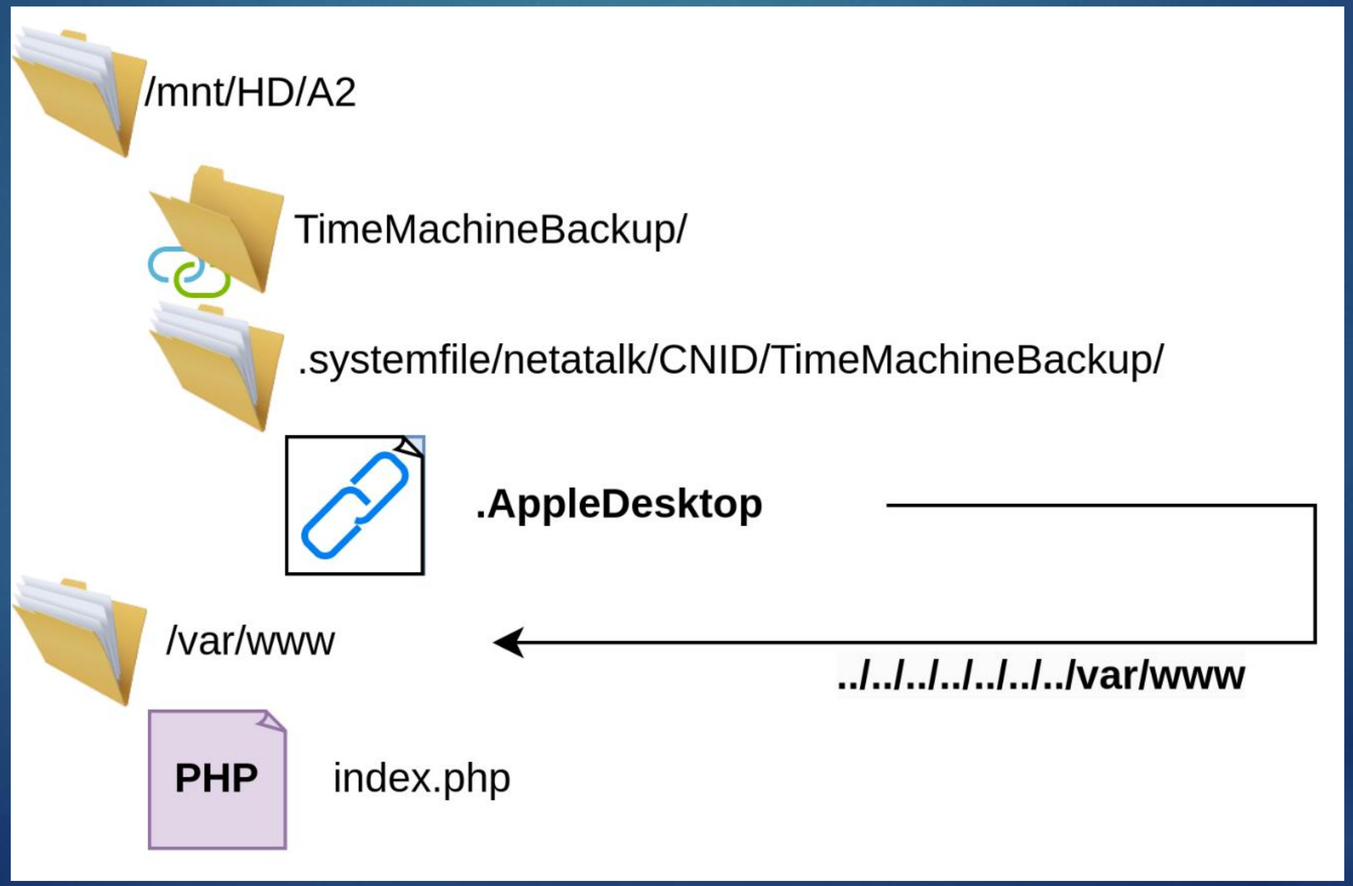
Exploit strategy

- Move it to the root of shared folder and rename it “.AppleDesktop” (Using SMB, yes this is doable)
  - From now can’t use this symlink since it points out of the shared directory



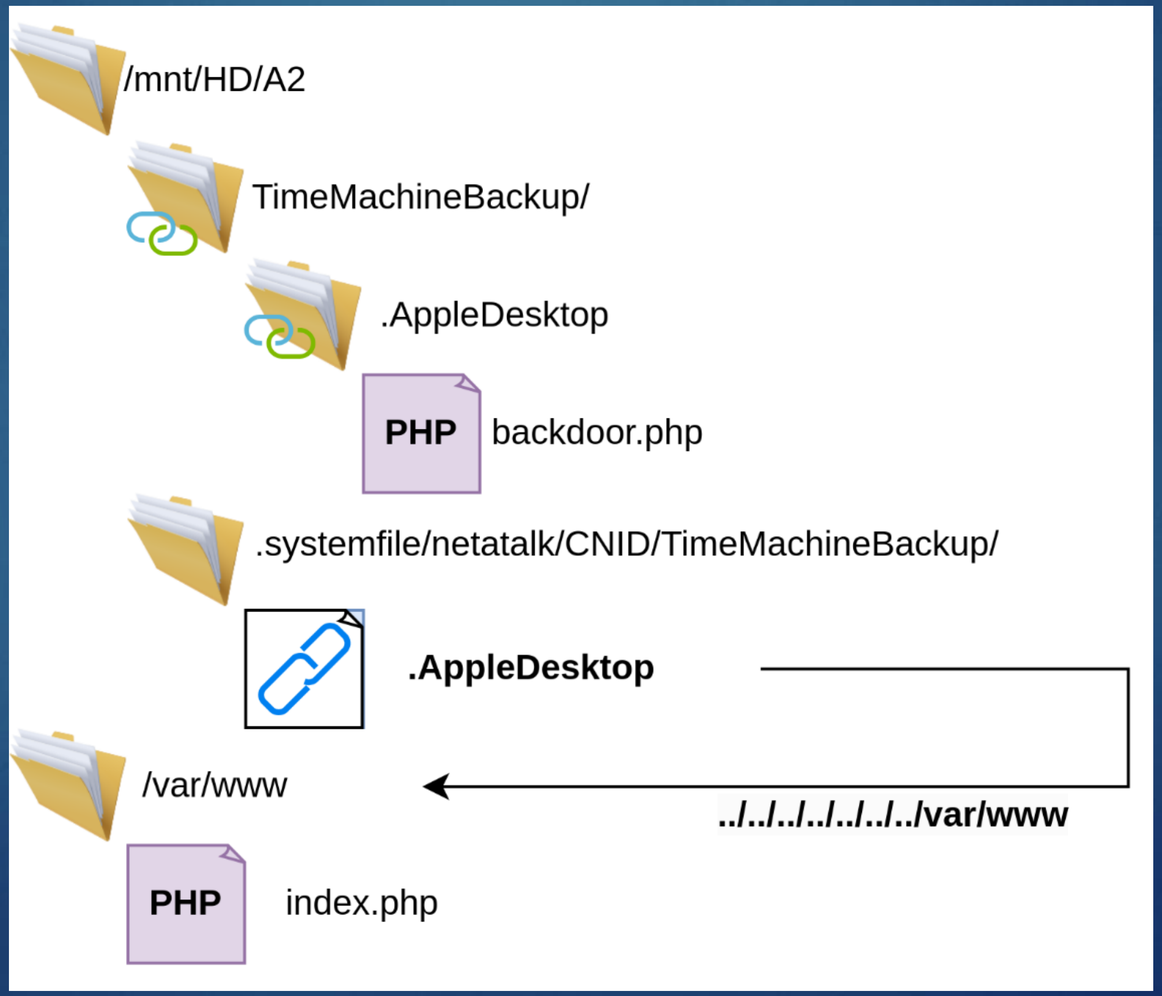
Exploit strategy

- Use create_apple_desktop to move the symlink to the DB folder
- **We now have a symlink in the DB folder that points to /var/www/**



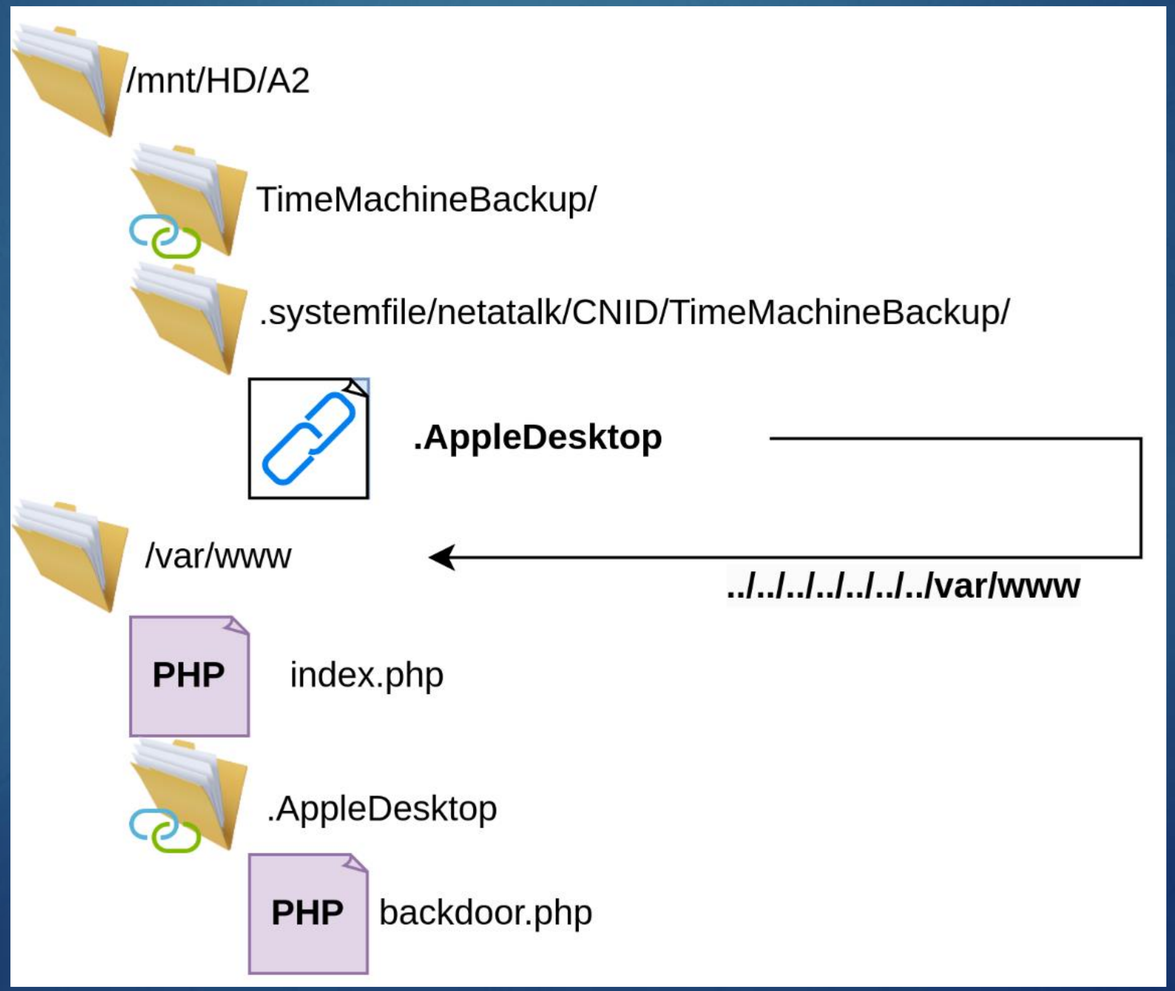
Exploit strategy

- Recreate a .AppleDesktop directory containing a PHP backdoor in the shared folder



Exploit strategy

- Reuse the create_apple_desktop to move a PHP backdoor to /var/www/
- **RCE**



Function create_apple_desktop

```
olddtpath = bfromcstr(vol->v_path);
bcatcstr(olddtpath, "/" ".AppleDesktop");

dtpath = bfromcstr(vol->v_dbpath);
bcatcstr(dtpath, "/" ".AppleDesktop");

if (lstat(cfrombstr(dtpath), &st) != 0) {

    become_root();

    if (lstat(cfrombstr(olddtpath), &st) == 0) {
        cmd_argv[0] = "mv";
        cmd_argv[1] = bdata(olddtpath);
        cmd_argv[2] = bdata(dtpath);
        cmd_argv[3] = NULL;
        if (run_cmd("mv", cmd_argv) != 0) {
            LOG(log_error, logtype_afpd, "moving .AppleDesktop from \"%s\" to \"%s\" failed",
                bdata(olddtpath), bdata(dtpath));
            mkdir(cfrombstr(dtpath), 0777);
        }
    } else {
        mkdir(cfrombstr(dtpath), 0777);
    }

    unbecome_root();
}
```

Exploit strategy

- **But requires a complex SMB / AFP setup and smart script to handle all this…**

Super_smart_sploit.py

```
# fixme
client.mkdir("%s" % random_dir)
client.mkdir("%s/B" % random_dir)
client.mkdir("%s/B/C" % random_dir)
client.mkdir("%s/B/C/D" % random_dir)
client.mkdir("%s/B/C/D/E" % random_dir)
client.mkdir("%s/B/C/D/E/F" % random_dir)
client.mkdir("%s/B/C/D/E/F/G" % random_dir)
client.mkdir("%s/B/C/D/E/F/G/H" % random_dir)
client.mkdir("%s/B/C/D/E/F/G/H/J" % random_dir)
client.mkdir("%s/B/C/D/E/F/G/H/J/I" % random_dir)
client.mkdir("%s/B/C/D/E/F/G/H/J/I/K" % random_dir)



temp = tempfile.NamedTemporaryFile()
temp.write(b"../../../../../../../../var/www/")
temp.flush()
```

Pwn2Own Contest

- Pwned both WesternDigital
  - One with this shitty ‘exploit’
  - One with a complex memory corruption requiring to rewrite a TCP stack
    - See Synacktiv blogpost
    - https://www.synacktiv.com/publications/exploiting-a-remote-heap-overflow-with-a-custom-tcp-stack
- Proceed to win the contest with other entries of the Synacktiv team
  - Master of Pwn

Conclusion

- The vulnerability is a bit hard to qualify
  - Misconfiguration between two services ?
  - **create_apple_desktop** bug ?
- Fixes:
  - **.AppleDesktop** is now forbidden by default in SMB (veto files)
  - **create_apple_desktop** was refactored
    - Checks properly that the source is a directory
    - Does not use bash commands to move the file…

Conclusion

- Pwn2Own is a super difficult contest
  - Very hard to win 40k $
- Still AFPD was shitty…

Conclusion

- New **PERMANENT** rule in all pwn2own contests since then:

Vulnerabilities in non-default apps/plugins, netatalk and MiniDLNA are out of scope.

Thank you !

See you at Cork in next week with same quality chains !

RE TACTICS

SYNACKTIV