Code Reuse Problem Set

Description

The goal of this problem set is to perform a code reuse attack against a vulnerable program, and develop a patch to remove the vulnerability.

To complete the problem set, you will need to ssh to your container at `$user@amplifier.ccs.neu.edu:$port`, where `$user` is your gitlab username and `$port` is your assigned ssh port (https://seclab-devel.ccs.neu.edu/snippets/6). Authentication is performed using any of your uploaded ssh public keys in gitlab.

You will also need to clone the problem set repository located at `git@seclab-devel.ccs.neu.edu:softvulnsec/prset04.git`.

Important Information	
Available	Mon 23 Mar 20:00 EST
Submission Deadline	Tue 31 Mar 18:00 EST
Gitlab URL	https://seclab-devel.ccs.neu.edu/softvulnsec/prset04.git (https://seclab-devel.ccs.neu.edu/softvulnsec/prset04.git)

Vulnerability Identification

A compiled version of the vulnerable program from `prset04.git` is located at `/usr/local/bin/prset04` on your container. While this program is easily exploitable, it is also protected by ASLR, DEP, and stack canaries.

Take advantage of an information disclosure vulnerability to leak address information from the program. Use this information to derandomize the address space layout and defeat stack protection.

Gadgets

The vulnerable program is running setuid 1001. Design an exploit that leaks the contents of `/usr/local/share/prset04.secret`, which is only readable by UID 1001.

However, since the program is protected by DEP, you will need to perform a code-reuse attack. Identify a set of gadgets that will implement your attack.

Obtain a Secret

Execute your attack to obtain the secret value.

Answer Submission

Fork the repository for this problem set in gitlab. Commit a JSON object to `solution.json` with the following format:

```
{
    "gadget_chain": [
        "<addr1 as 0xhex>",
        "<addr2 as 0xhex>",
        // ...
    ],
    "secret": "<secret value>"
}
```

For example, given a gadget chain of `0x1234` → `0x5678` and secret `abcd`:

```
{
    "gadget_chain": [
        "0x1234",
        "0x5678"
    ],
    "secret": "abcd"
}
```

You are responsible for submitting valid JSON at the correct path. Use a validator if you're unsure about this, and double-check that your JSON follows the format above exactly.

In addition, commit your exploit to `exploit/` and a `README.md` that describes your exploit *as precisely as possible*.

Updated Fri 07 Aug 2015 10:38 EDT
Revision master/52d826a

bootstrap (http://getbootstrap.com/) — ember (http://embe