Trying to make it OverTheWire

Erik Pedersen - z5311210

November 2, 2024

Link to the wargames writeups: `https://github.com/erik-pedersen/COMP6841`

If that link does not work, here are links to the HTML versions of the writeups:

- My student page

  `https://cgi.cse.unsw.edu.au/~z5311210/`

- This report
- Natas writeup
- Leviathan writeup
- Krypton writeup
- Narnia writeup
- Behemoth writeup

Key things learnt:

- What the hell PHP is (natas)
- Basics of pwntools, gdb/pwndbg
- ltrace and strace (so helpful! Most of Leviathan)
- A little bit of x86 assembly (gdb)
- Some other examples of where buffer overflow can be used (narnia)
- Some cool techniques e.g 'yes/no querying' (Natas15 → Natas16)
- XOR encryption (natas11)
- An alternative buffer overflow attack involving environment variables (Behemoth1)
- Exploiting system() calls by manipulating $PATH (Behemoth2)

- Basic cryptography (krypton)

Resources used:

- Google, mostly lol
  - Like seriously, when learning new techniques/tools, I was mostly just Googling how to do things.
- I did some narnia, and found https://shell-storm.org/shellcode/index.html quite helpful
- Online PHP interpreter
- For Krypton and Natas, a bunch of online tools for hashing, encoding/decoding things.

Mindset changes:

- Do not be afraid of looking at solutions. Sometimes you genuinely do not know.
- Probing whatever you're trying to find an exploit can be very fruitful. Instead of rushing into doing technical things, just see if you can break the program with some weird inputs!
- Don't be afraid to FAAFO (Mess around and find out) in the war games. I spent quite a bit of time just messing around with natas15 to see what I could do (that's where webshell.sh came from!) - it really helps you better understand what's going on, and might give some ideas for the future.
- Using previously learnt tools in future tools: I learnt how to use ltrace in leviathan, and used it again just as a basic tool in narnia2.

Tools I made:

- I'm sure some things could be adapted to be more modular, they're all mostly for a particular war game at the moment.
- That being said, I have become better at actually writing tools to solve problems
- get_addr: Small C program that, given the name of the target file and name of environment variable, tells you the address that environment variable will be located.
- Some python scripts for encoding/decoding in krypton
- A set of similar tools for slowly peacing together flags in Natas
- A series of tools for piping shellcode into binaries (the pwnarniaX.py series :D)

What I want to do next:

- Finish off Narnia and Behemoth
- Try out some more hackthebox :)

Advice to my past self:

- Manage time better.

Your security engineering understanding and growth over the duration of the project:

- I have a much better idea of the types of (arguably, quite legacy) vulnerabilities that exist in binarys (Narnia & Behemoth), and some web vulnerabilities (Natas). I also did a little bit of cryptography stuff in Krypton, but honestly found it quite boring.

Personal analysis and reflections that make your journey and/or resources unique:

- All of my analysis was completed as part of the corresponding war game folders.

Analysis: The level of depth you have explored in your chosen project:

- I wish I spent more time learning how to do the wargames in Behemoth. Although I made some progress, it looks really fun. Instead, I spent a considerable amount of time doing Natas. Atleast that gave me some experience writing web-interacting Python!

Reflection: How you document issues you encounter and how you overcame that adversity:

- Any issues I had are also part of the analysis, things I tried that did not work, times where I was not making any progress.

Progression of your project proposal

- I do not think I met the expectation I outlined in the project proposal. I think I got too carried away doing Natas that I neglected the other war games. Again, I wish I did more Behemoth lol

Impressive elements of your project - what cool stuff did you do?!:

- I think the coolest part of this project was when I was able to draw on knowledge I learnt about in previous CTFs to use in future CTFs. My first instinct whenever I ran into a new binary or website is to play around with it in weird ways, for binarys I run ltrace/strace, for websites I inspect element.

- I'd also find myself drawing on previous knowledge to try and crack a problem. My first instinct in Behemoth1 was to write shellcode into a buffer and then return back to that buffer. That did not end up working - and I'm glad it didn't, because it meant I got to learn more about environment variables!
- Another cool and satisfying part is all the little programs that I made along the way to help solve the war games. Although they aren't very elegant, they work, and for that I am proud :)

Link to the wargames writeups:
`https://github.com/erik-pedersen/COMP6841`
The writeups can be found under 'project/[WARGAME_NAME]/README.md'
Thanks for reading :)