Events

HXP CTF - Credentials in Slack - Friday, November 17th 6 AM to Sunday, November 19th, 6AM

Officer Applications - Open Monday
CTF Topology

- Auxiliary Network Services
- CTF Server
- Challenge Server(s)
Key Considerations

- Reliability
- Integrity
How do you maintain reliable, solvable challenges on a hostile network?
Auxiliary Network Services

- DHCP
- DNS
Fun with DNS

generichax.ctf

vs.

generichax.csg.utdallas.edu
CTF Server

- Hosts the CTF Challenges
- Generally CTFd or a custom solution
Challenge Server

- How do you host vulnerable services in a secure fashion?
- Again, reliability and integrity
- Someone shouldn’t be able to use challenge A to solve challenge B
Separation Techniques

- Physical
- Hypervisor
- Container
Physical Separation

- Physical security
- Requires lots of machines
- Requires lots of power
- Requires lots of management
- :(  


Hypervisor Separation

- Hypervisor level security
- Requires 1 good machine
- Extra management overhead
- Low utilization
- Compounding problems if doing service replication
- Recommendation: vSphere or Proxmox
Container Separation

- Kernel level security
- Can be run on physical machines or VMs
- Environment reproducibility
- High utilization
- Recommendation: Docker Swarm
My proposed architecture

- PFSense gateway - provides DHCP, DNS, etc.
- CentOS - managed by Puppet
- CTFd and Challenges - Docker Swarm
- NGINX Load Balancer for everything
Monitoring

- Verify challenge can be reached
- Verify challenge can be solved
- Predict future needs
CTF Challenge Creation

- How do I guarantee my challenge will work when deployed?
- How do I guarantee my challenge is running?
- How do I guarantee my challenge is solvable?
Docker Containers

- Integrate the challenge into the deployment environment
- Built in health check to verify the challenge is running
- Built in health check to verify challenge is solvable