# King of the Hill

## A Novel Cybersecurity Competition for Teaching Penetration Testing


Kevin Bock, George Hughey, Dave Levin

# Teaching Penetration Testing

- Businesses are increasingly ramping up internal security
- Penetration testing has exploded as a field
- Teaching pentesting has become increasingly in-demand

# Teaching Cybersecurity

- Cybersecurity competitions are a effective and engaging way to students to learn and practice cybersecurity
- Many different types of competitions geared to teaching different aspects of security



# Penetration Testing

**Pivoting**
from one machine to another

**Implants**
developed in advance for an engagement

**Preparation**
with advanced recon, scanning, and development

# Goals

1. Require competitors to **pivot**
2. Allow for the development of **implants**
3. Allow for advance **reconnaissance** before the competition
4. Encourage defensive operations and **trade-offs**
5. Instill **best practices** for both offense and defense, and keep **ethics** in mind

# King of the Hill (KotH)

# KotH at a High Level

**Pivoting** — Large, nontrivial **network topology** with pivot points

**Implants** — Student **teams** write offensive and defensive implants

**Preparation** — **Two-week project**: find vulnerabilities & write implants

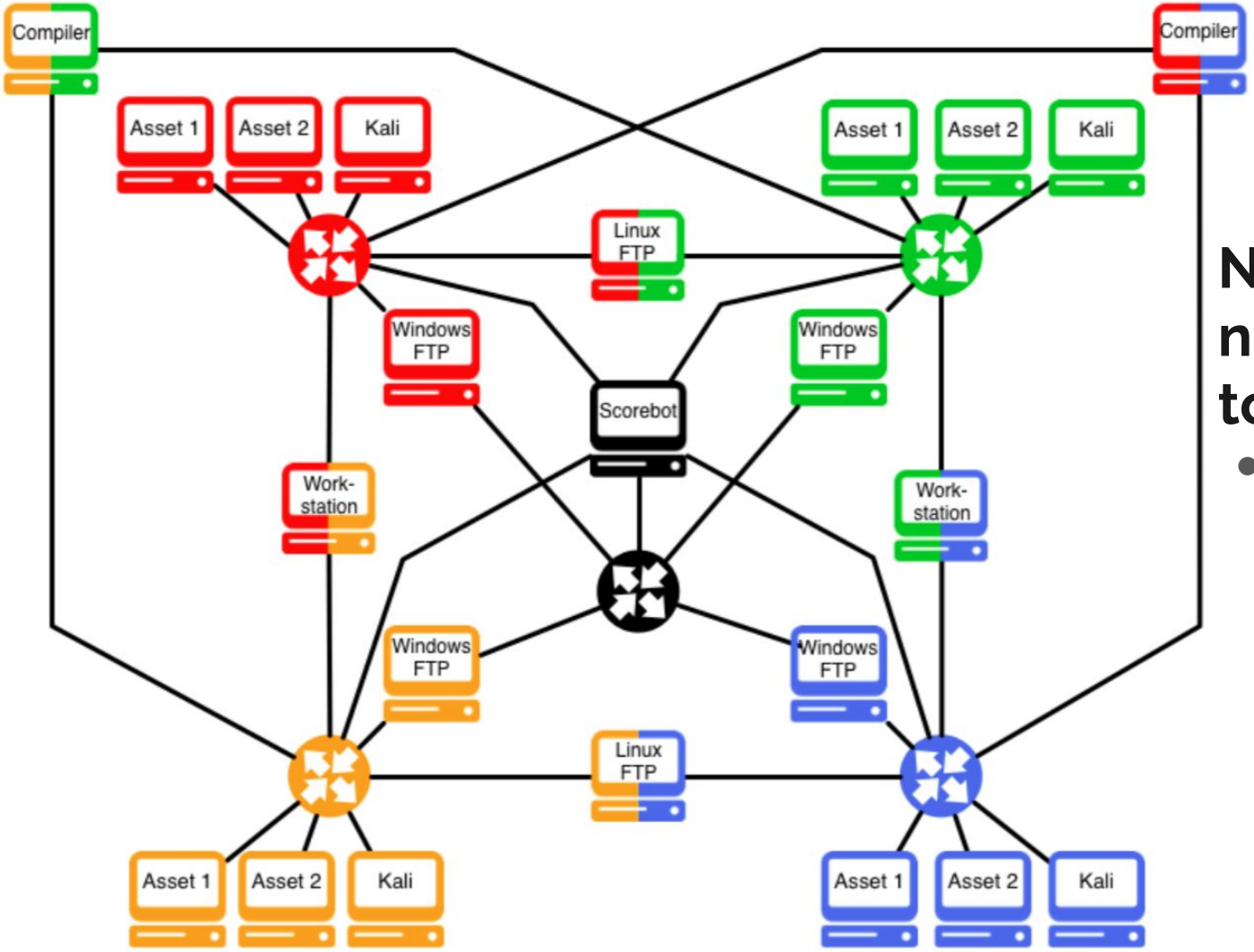
**Trade-offs** — In-class competition, decide what **critical services** to defend, patch, or turn off

# Gameplay

- Students divided into teams
- Each team must work together to *attack*, *control*, and *defend* machines over a large network topology

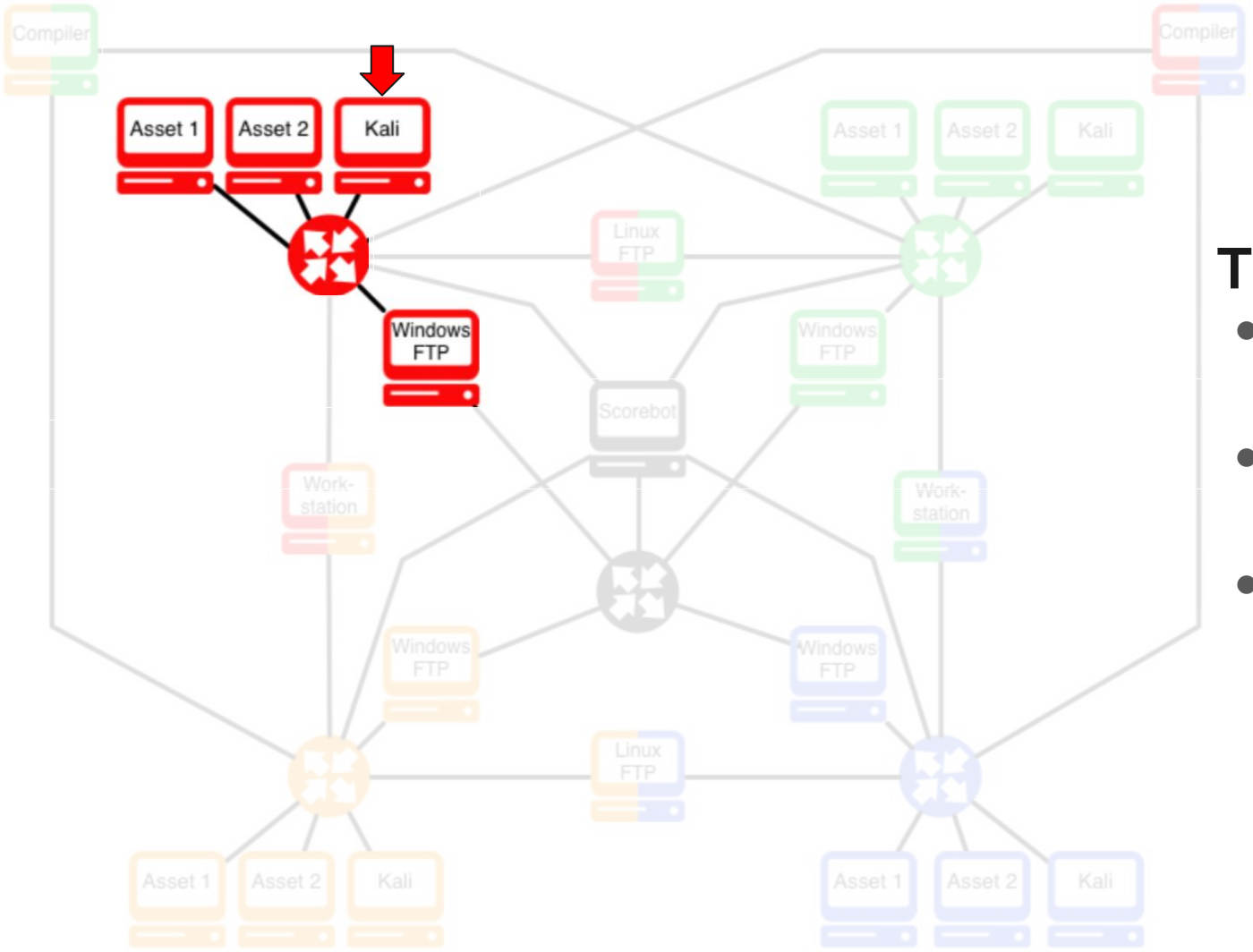
# Maintaining Critical Services

- As students claim machines, they inherit the *responsibility* for them
  - Must protect their access and critical services from other teams
- We've introduced vulnerabilities
  - Competitors face a **trade-off**: patch or turn off?
- **Scoring**: Every two minutes we check for service availability
  - +1 point for each machine they control that is up/responsive to pings
  - +1 point for each functioning critical service



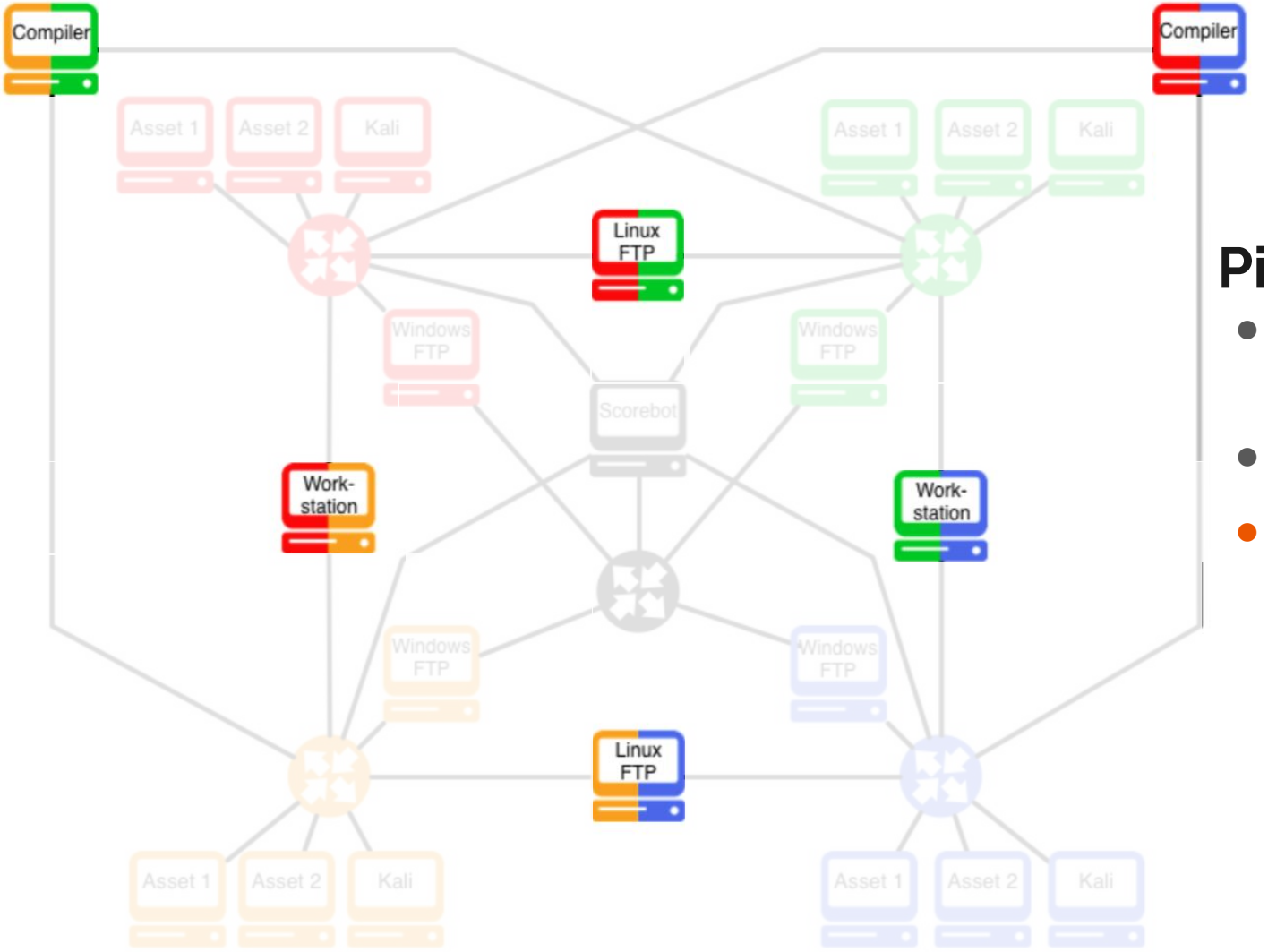

# Nontrivial network topology

- Variety of different machines, operating systems, and services



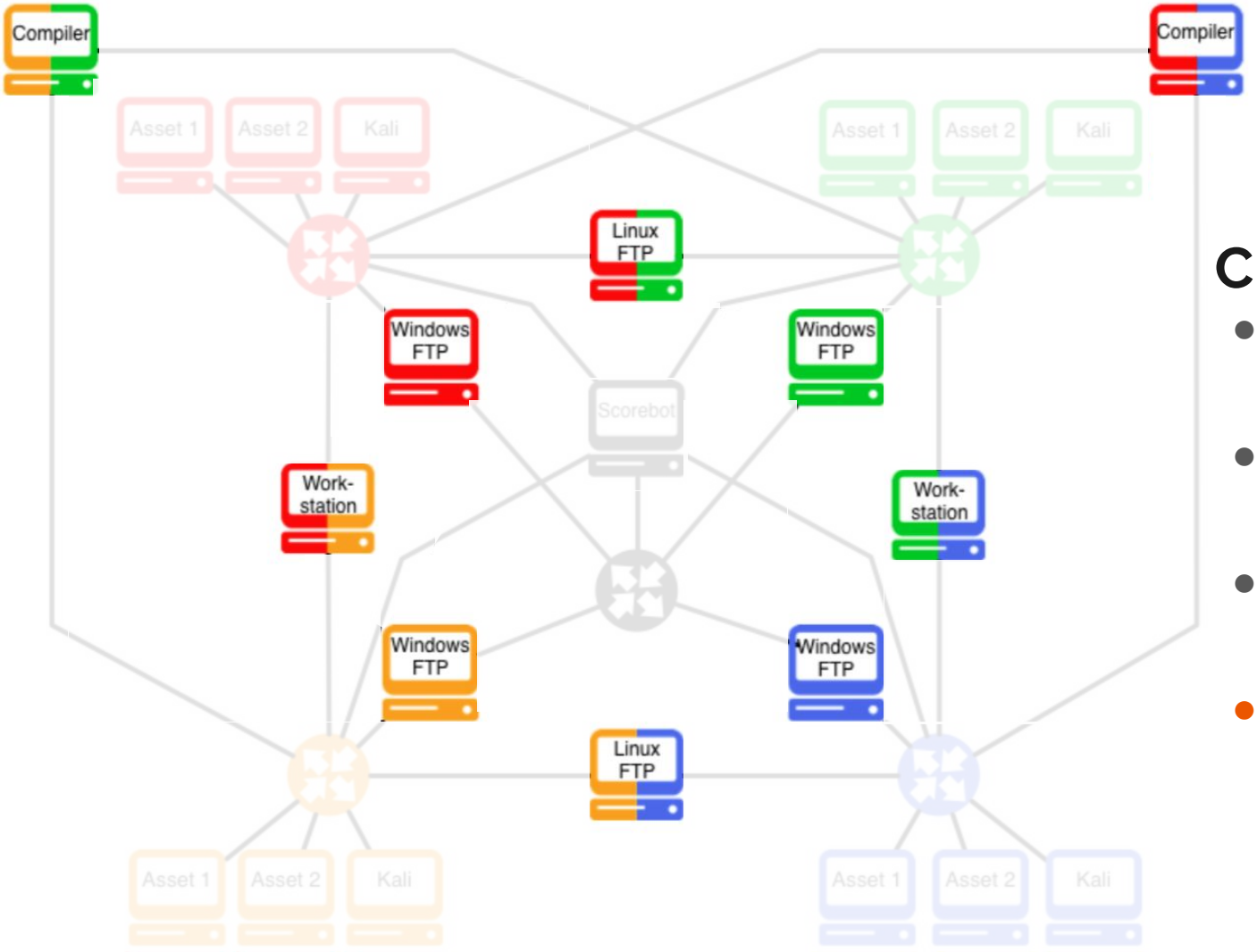

# Territories

- Each team starts with an entry node
- Entry machines are *out of scope*
- Territories grow/shrink as teams take/lose control of boxes



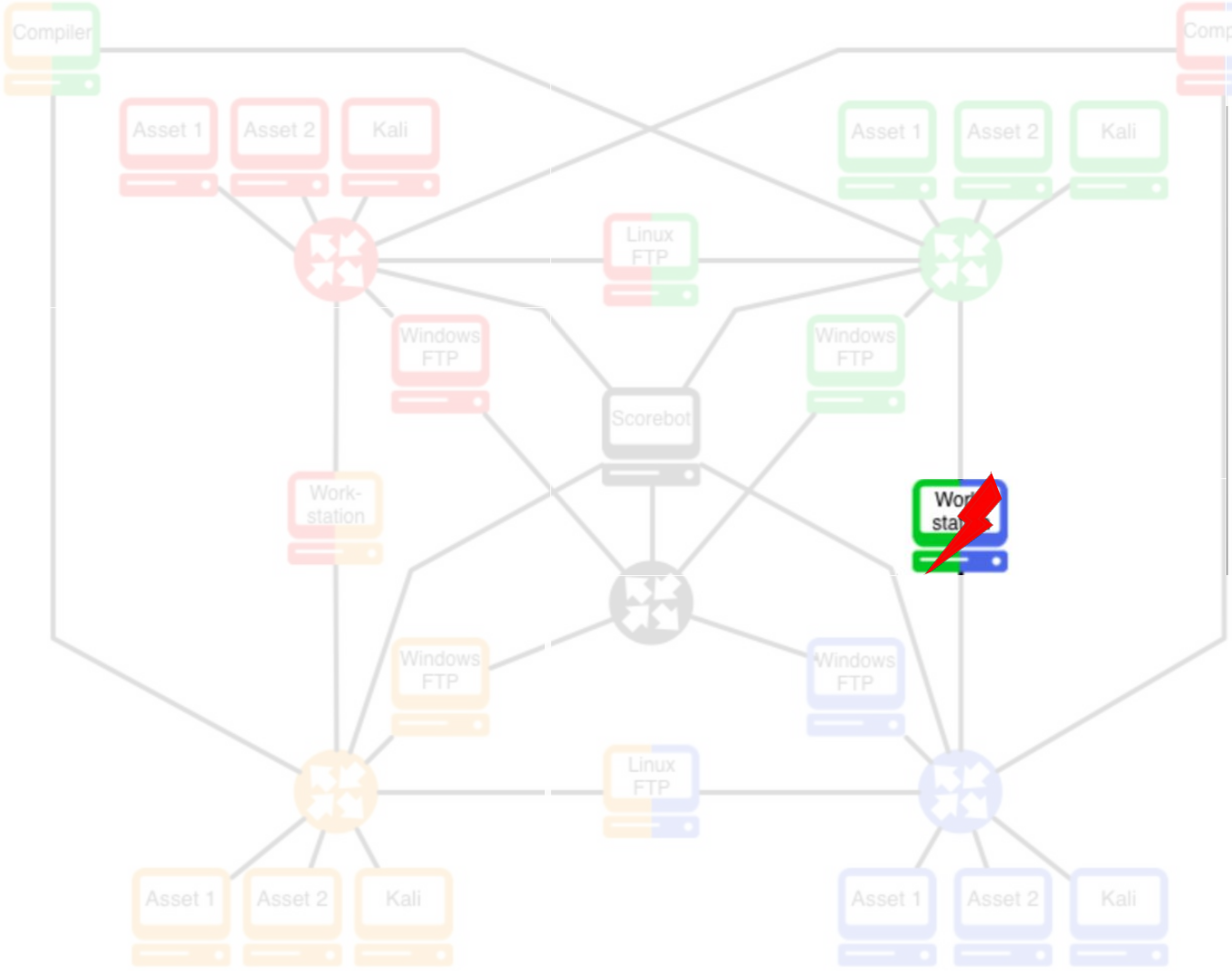

# Pivots

- Necessary to access other subnets
- High-value targets
- We expect these to change hands many times throughout a competition



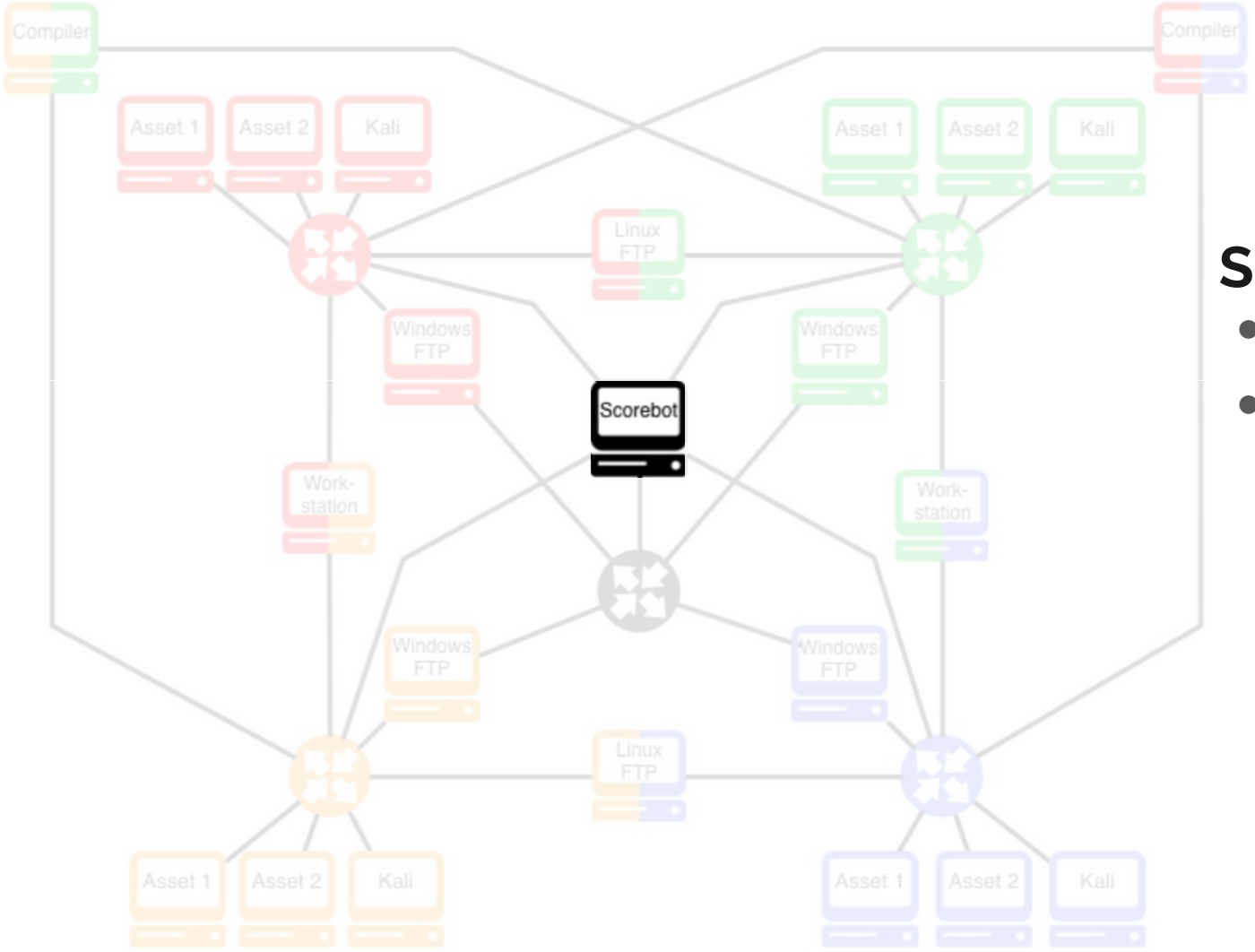

# Critical services

- FTP, SSH, HTTP, etc.
- Must be maintained and protected
- Pre-seeded vulnerabilities
- We expect these to become *more secure* throughout a competition



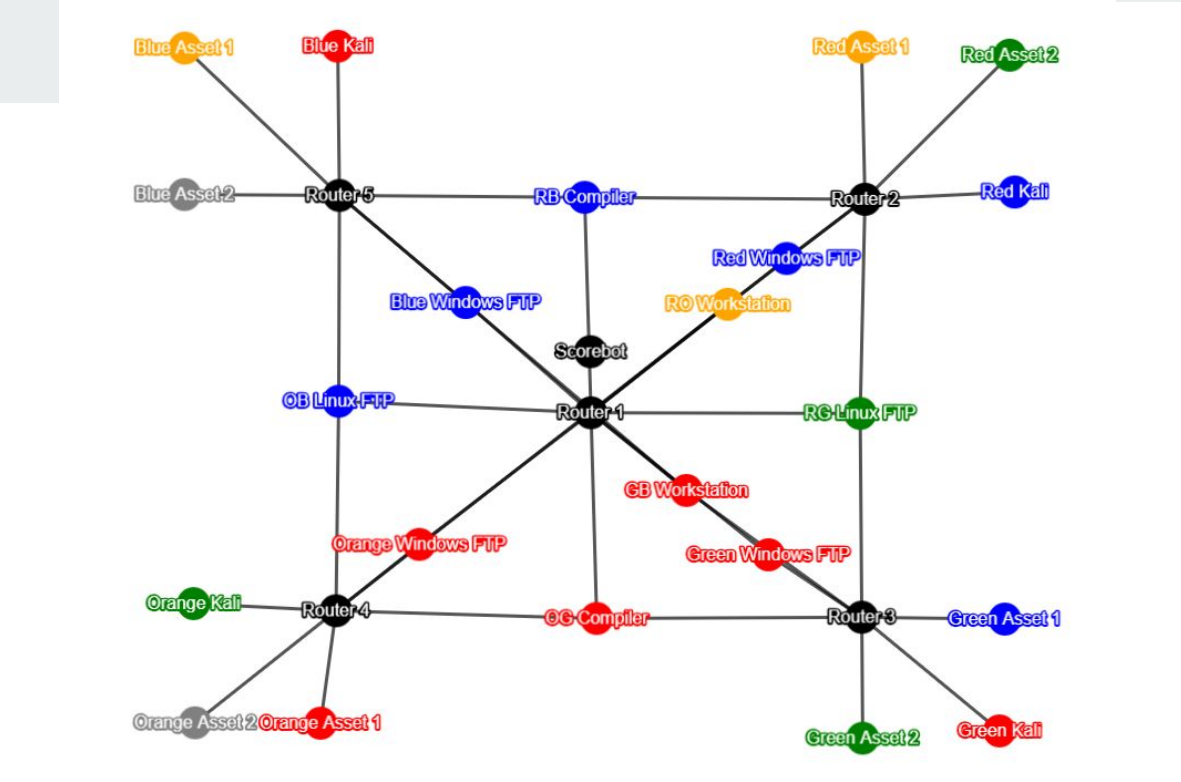

# Implants

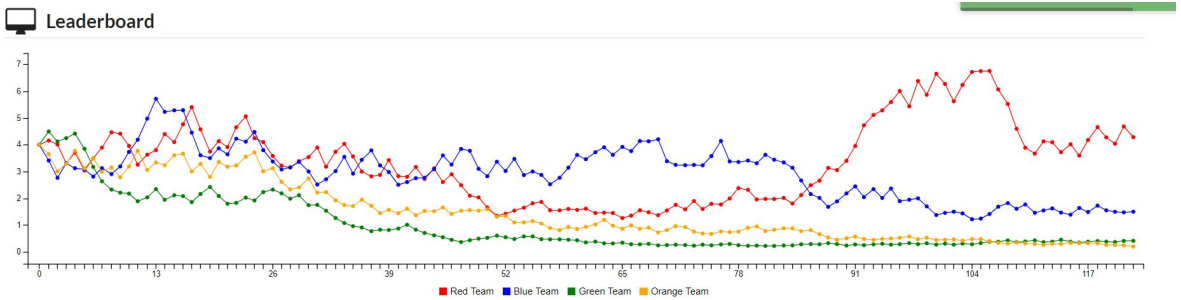
- Teams develop implants in advance
- We deploy them on target machines
- No teams know what or where other teams' implants are




# Scorebot

- Globally reachable
- Periodically verifies critical services are responsive




# Scorebot dashboard

- Accessible by all teams
- Updates live as teams claim machines
- Shows where attacks are taking place
- Shows the overall accessibility of each service

# Student Preparation

- Each team is given a full, isolated clone of the competition environment 2 weeks in advance of live competition to privately penetration test the network
- Each team could enter the competition with overlapping but different ways to access, escalate privileges, and defend different target machines

In our class, each student **identified 2 vulnerabilities** on an image of their choice and **wrote an implant** as a project.

# Continual Scanning

- During the competition, a few highly vulnerable, unscored, hidden machines are secretly added to the network that do not appear in the initial network copies
- Easy to breach compared to the rest of the network
- Pose a threat to teams if other teams can attack them through previously unseen vectors
- Mimics threats faced often by real Network Operation Centers of new vulnerable or compromised machines being connected by unknowing employees, insider threats, or malicious actors

# KotH at a High Level

**Pivoting** — Large, nontrivial **network topology** with pivot points

**Implants** — Student **teams** write offensive and defensive implants

**Preparation** — **Two-week project**: find vulnerabilities & write implants

**Trade-offs** — In-class competition, decide what **critical services** to defend, patch, or turn off

# Implementation

- Competition backend was designed and run in Cypherpath
  - Virtual Software Defined Infrastructure (SDI) management program

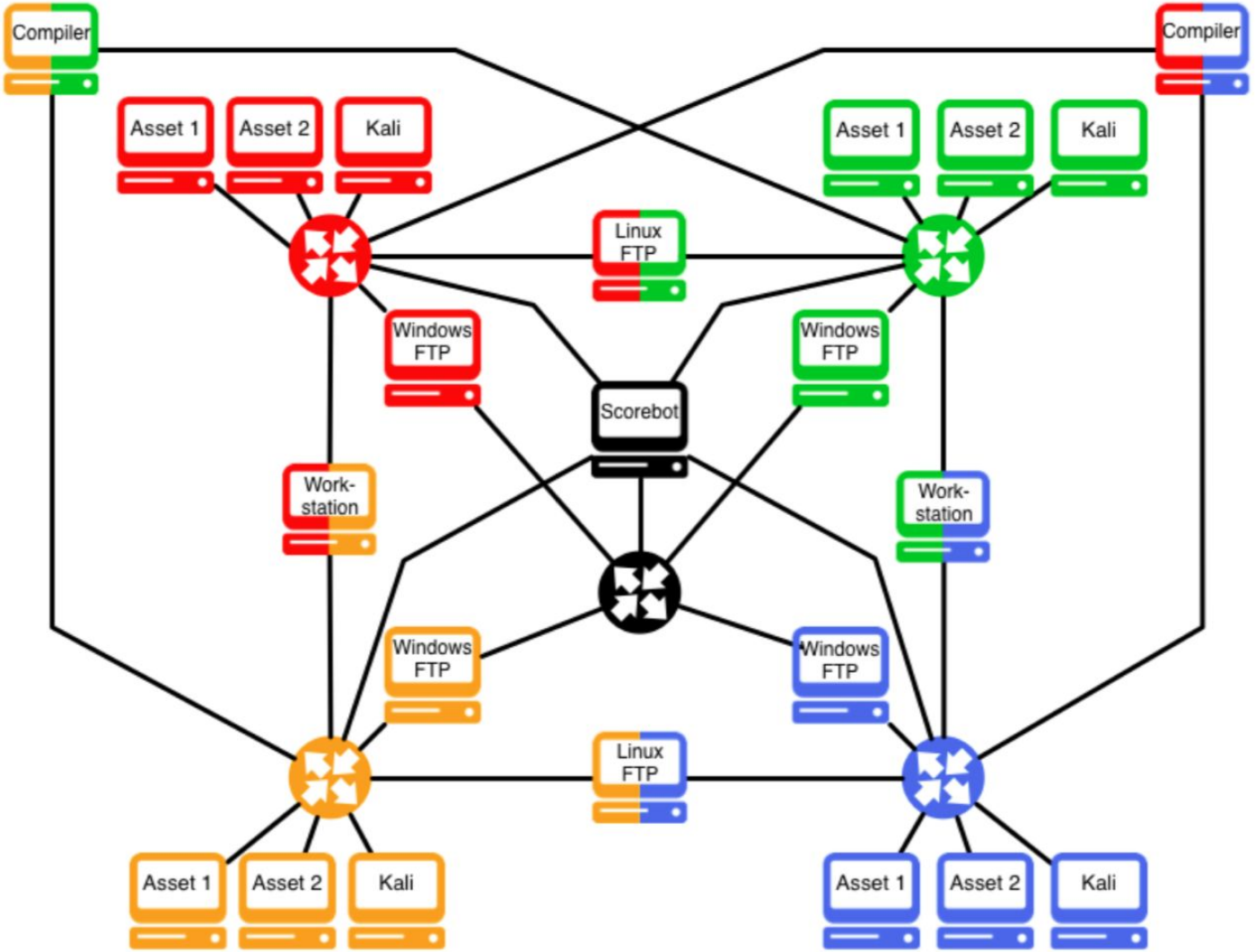
- The network layout, machine information, and scorebot implementation are publicly available

https://koth.cs.umd.edu

# Sample Run

# Sample Run

- Ran King of the Hill for our course on *Introduction to Penetration Testing*
  - Exercise ran for 3 hours
- Configuration:
  - 4 teams of 4-5 people, labeled by color
  - Each team got an initial Kali machine reachable only by them
  - Six unique vulnerable images (4 Linux, 2 Windows)
    - Duplicated them across the networks
    - Every team's view was symmetrical


Compiler
Compiler
Asset 1
Asset 2
Kali
Asset 1
Asset 2
Kali
Linux FTP
Windows FTP
Windows FTP
Scorebot
Work-station
Work-station
Windows FTP
Windows FTP
Linux FTP
Asset 1
Asset 2
Kali
Asset 1
Asset 2
Kali

# Results

- Students were quick to close vulnerabilities after gaining access
- Worked to configure strong firewall policies to block traffic on unwanted ports
- Carefully monitored running services and processes to find malicious or vulnerable code
- By the end of the competition, most machines were **significantly more secure** than at the beginning

# Cost-Benefit

- Students identified some vulnerabilities that were more time-consuming to patch and chose to *leave them unpatched*
- Weighted cost of lost points during patching against the risk of another team exploiting them
- Multiple teams specifically reported this for EternalBlue

# Vulnerability Discovery

- Across all teams, students identified most access vulnerabilities
- Local privilege escalation (LPE) vulnerabilities were most often missed by students during initial penetration test
- Only unprivileged access is required to trigger a phone-home to the scorebot to claim a machine
    - Privileged access is primarily useful for bolstering access and acting defensively

# Vulnerability Discovery

- Before the competition, students valued unprivileged access more than a full-chain of exploits (access + LPE)
- This dynamic changed during the competition
- Many machines had multiple teams simultaneously accessing them with unprivileged access
    - became a “race to root” of which team could escalate their privileges to kick out the other teams first and defend the machine

# Implants

- Students put a great deal of effort in implant development
- Many very strong implants were developed
  - Recompiled Bash with backdoors introduced
  - Infected/hooked PAM module
  - Self-hiding and self-protecting userland rootkits
  - Small kernel module
  - Self-protecting backdoor processes

# Student Feedback

- Student feedback was overall very positive
- Students became very invested in the competition, and worked hard on implants and vulnerability discovery
- Students liked the dual attack-defense nature coupled with the ability to strategize

# Customize KotH for Your Class

| | |
|---|---|
| **Pivoting** | Specific vulnerabilities can vary based on class goals<br>Network topology can establish attack “prerequisites” |
| **Implants** | Class projects could require certain attacks/defenses |
| **Preparation** | Varying amount of details can be provided |
| **Trade-offs** | Patch vs. turn off; easy vs. hard targets; attack vs. defend |

# Summary

- King of the Hill is a novel cybersecurity competition that provides hands-on experience with real-world penetration testing practices
- Combines
  - network pivoting
  - custom implant development
  - advanced preparation
- Initial in-class run of KotH indicates that it creates an exciting environment in which students gain valuable pentesting practice

https://koth.cs.umd.edu

# King of the Hill