Pros vs. Joes CTF Player's Guide

Errata

Mistakes happen. Please send errata to daniel@planethacker.net and captainopsec@prosversusjoes.net

Shout out: Credit and a special thanks to dmfr (daniel@planethehacker.net) for the initial draft of this document. Various contributors of the Pros Versus Joes Tribe have modified his original work to adapt it for a virtual game. This document will evolve to improve the Pros V Joes CTF player experience.

What to expect

This CTF is an attack/defend event. Blue teams are assigned identical networks that have been compromised by a red team. The blue teams' objectives are to maintain availability of scored services and to eradicate the Red Team's persistence mechanisms. Typically, the blue teams will focus only on defense for the first half of the game. During the second half, they may employ offensive techniques against other blue teams. The final stretch of the game is known as "Scorched Earth", and destructive attacks are permitted insofar as the Pros Versus Joes CTF Rules allow.

To prove that your team has compromised another team's assets, you must send a beacon to a scoring server containing a token assigned to your team. Services are scored by availability; if your services are functioning properly and accessible to the scoring server, points will be awarded. If the services are not functioning as intended, points will not be awarded. Services are checked and points are awarded in roughly 5 minute increments.

Rules may vary per event. Please make sure to read and understand the rules prior to playing. Points may be deducted for rule infractions or behavior that is deemed to be unsportsmanlike. Bonus points may be awarded for clever maneuvers at the discretion of the Gold Team.

The systems you will defend are a variety of Windows, Linux, and possibly other operating systems. You may not know the network's composition until the day of the

event, so plan to be able to deal with a variety of operating systems at different versions, patch levels, and installed software.

You will usually have a few weeks ahead of an event to prepare. Team captains are responsible for preparing their teams to succeed at the event. Successful teams have the majority of players dedicating an hour or more per day preparing to play in the days leading up to the event.

Before COVID-19, Blue teams were required to play in-person. As of 2020, the game has been adapted to allow for a virtual format. Pros V Joes, in partnership with B-Sides staff, will determine the format for each game.

For In-Person Games: Player hardware must be able to connect to a wired network and use VPN software to access the range. It is a good idea to use a checklist to make sure you do not forget key items such as Ethernet cables and dongles. You will typically NOT be able to use WiFi to access the range. These are long days, so remember to pack any food, drinks, medication, and snacks you may require. Space is also limited (teams of ~10 typically share two folding tables), so take this into consideration as well.

Common Pitfalls

Do not let any of this advice discourage you from attempting anything on this list. They are all good ideas, but have caveats that may not be obvious. All of these items listed below have

Keep in mind that the environment provided to you will likely be unknown to your team until game day. You will not have luxuries you may have at work or home at your disposal such as the ability to snapshot or revert VMs, rebuilding systems from scratch, or even having the correct credentials to login. You probably won't have a functioning SIEM or even basic centralized logging. The systems will be running a mixed bag of software that you may have little or no experience with. Some systems are likely to be EOL and have broken package managers. In short, expect things NOT to go smoothly.

There will also be a limited time available to work. Depending on the skills and abilities of your team, it may make sense to make unorthodox choices. If you think it will take an entire day to set up centralized logging on roughly a dozen systems in a heterogeneous environment, it may not be worth it even though it is industry standard best practices to have such systems in place. If you decide to use automation such as Ansible to apply hardening playbooks, realize that you will be working with systems that might not have Python installed and it may not be very easy to get it installed. Dealing with the

unexpected and making smart choices when triaging systems is one of the keys to victory in this game.

Here are a list of common good ideas and assumptions that **may not work very well in the context of this event**:

- Just firewall everything! Teams must allow all traffic to scored services and may not block anything by source IP. Since you will not know what you are up against, making generic firewall rules may not be practical. Overly-restrictive firewall rules are actively hunted out and penalized by the Red and Gold teams. People deploying firewall rules must be able to troubleshoot their firewall rules and not expect someone else on the team to do it for them. Several teams have been overzealous with firewall rules and taken down their entire infrastructure for extended periods of time.
- Configuration management/automation for hardening. Again, you will not know what Operating systems, distributions, or software to expect. Your playbooks, DSCs, manifests, etc. will probably not work on everything unless you spend a lot of time during the preparation phase to account for all of the possible differences. People working on automation must be able to troubleshoot their tools and not place this burden on other team mates. Several teams have deployed hardening that broke systems or locked them out, requiring a lot of time to be wasted trying to rescue these systems and points spent on reverts.
- Taking snapshots or reverting. Snapshots are typically not available for this game. Reverts are available at a cost to points. Reverts are also performed by the Gold Team at their convenience. If you break a system, it may be several minutes before you are able to revert it. Several teams have been bitten by the assumption that they can take snapshots and perform reverts at will.
- Rebuilding systems from scratch. This is possible, but you must purchase a system from the store at a points penalty. it may take a few hours to install and configure the required software and lack key configurations or data required for scoring. The time required to pull this off may not be worth the sacrifice in points or resources.
- Using tooling that only one person on the team knows how to use and administer effectively. This makes you effectively lose that player for the duration of them setting the tool up, and it may not provide a good return on investment. Furthermore, if this player leaves to go eat lunch or needs a break, this tool is effectively dead in the water. Choose these technologies carefully before committing to them.
- Full disk/memory forensic images. Transferring these over the network may take too long.
- Developing tools that require dependencies. It may be very difficult to install libraries required to make custom tooling function correctly. If possible, bring

precompiled, static-linked binaries or programs/scripts that don't require libraries or modules. Several teams have developed custom tools, then couldn't get them to work on the systems they needed them on due to the inability to install the required libraries, or an incorrect version of a script interpreter installed (requires Python3, but Python2.5 is installed. System is so old Python3 isn't in packages and the required libraries to compile Python3 aren't available in the package manager).
- Running vuln scanners/discovery tools against other teams. It is effective to run scanners against your own stuff, but for discovery, you can just login to your team's equivalent server and look for yourself. Also, the other teams will be vulnerable to the same things you are so you can use your vulnerability scan results against their stuff.

Strategies That Work

Keeping it simple, doing your homework, and making calculated, well thought out decisions goes a long way. Here are some ideas that winning teams have employed.

- Make checklists. It is easy to skip or forget to do key tasks when hardening or hunting on an endpoint. All mission critical jobs have checklists; airline pilots and astronauts use them before takeoff, soldiers use them before deploying or embarking on a mission, etc. Checklists are especially useful when dealing with stressful or chaotic situations.
- Actually use your preparation time to prepare! There is a lot of information out there that reveals information about what to expect, what worked and didn't work, and even information about the systems you may be defending. Getting practice time in with the tools you will be using helps a lot.
- Committing to your role. If you are in charge of the DNS server or firewalls, that's what you should focus on rather than bouncing around from system to system. Most people are unable to switch contexts in short amounts of time effectively.
- Communication. Tell people what you are working on so efforts aren't duplicated. Let someone know if you are stuck. Get to know your teammates ahead of time! Some people work really well in pairs, others are better solo. Organize your chat channels so Linux admins aren't getting notifications for Windows systems and vice versa to reduce alert fatigue.
- Making appropriate technical choices; focus on things that are generically easy to deploy, fast, and effective. Try to achieve as much coverage as possible over a framework such as MITRE ATT&CK to increase your chances of catching malware and persistence. Beat up offensive players with a solid mastery of defensive basics.