Deceiving the Deceivers: A Review of Deception Pro

Published: 2026-01-13 · Archived: 2026-04-05 18:10:22 UTC

TL;DR: *This is my personal experience and a quick review of the Decepton Pro sandbox. Deception Pro is a specialized sandbox for long-duration analysis of malware and threat actor behavior. Overall, it is a promising product and fills a niche gap in the malware sandbox market.*

One challenge facing malware analysts, reverse engineers, and threat intelligence analysts is understanding how malware behaves over a longer period of time. Capturing behaviour in a traditional sandbox for 3, 5, or even 20 minutes is possible, and analysts can also run samples in custom virtual machines or a baremetal analysis system to watch what they do. But there are still key challenges, such as:

- It's difficult to make the environment realistic enough to "convince" malware to continue execution, and even more difficult to capture follow-on actions such as commands issued to a bot or hands-on-keyboard events. Advanced malware and actors are looking for active directory environments or corporate networks, for example, and this can be difficult to simulate or maintain.
- Even if an analyst can create a realistic enough environment to capture meaningful actor activity, it's difficult to randomize the environment enough to not be fingerprinted. If an actor sees the same hostname, IP, or environment configurations over and over again, the analysis machine can easily be tracked and/or blocklisted.
- Scalability, especially in baremetal setups, is always an issue. In my baremetal analysis workstation, I can't detonate multiple malware samples at a time (while preventing cross-contamination), for example, and I can't easily add snapshots for reverting after detonation.

Introducing Deception Pro

I was introduced to Deception Pro by a colleague who spoke highly of Paul Burbage's team and the work they've done on other products (like Malbeacon). After reaching out to Paul, he was kind enough to offer me a demo account to help me understand the product and how it could fit into my threat research workflow. So without further ado, here's my disclaimer:

Disclaimer: *Paul and the Deception Pro team provided me with a free demo license to evaluate the product and see if it meets my needs. I'm not being paid for this review, and Paul and the team did not ask me to write one. This review is entirely my own doing.*

In this post, I'll be covering what Deception Pro is, how it can fit into a malware analysis and reverse engineering workflow, and some of its features.

Overview

Deception Pro is what I'd call a "long-term observability sandbox." Essentially, it's a malware sandbox designed to run malware for extended periods – several hours or even days – while also fooling the malware into thinking

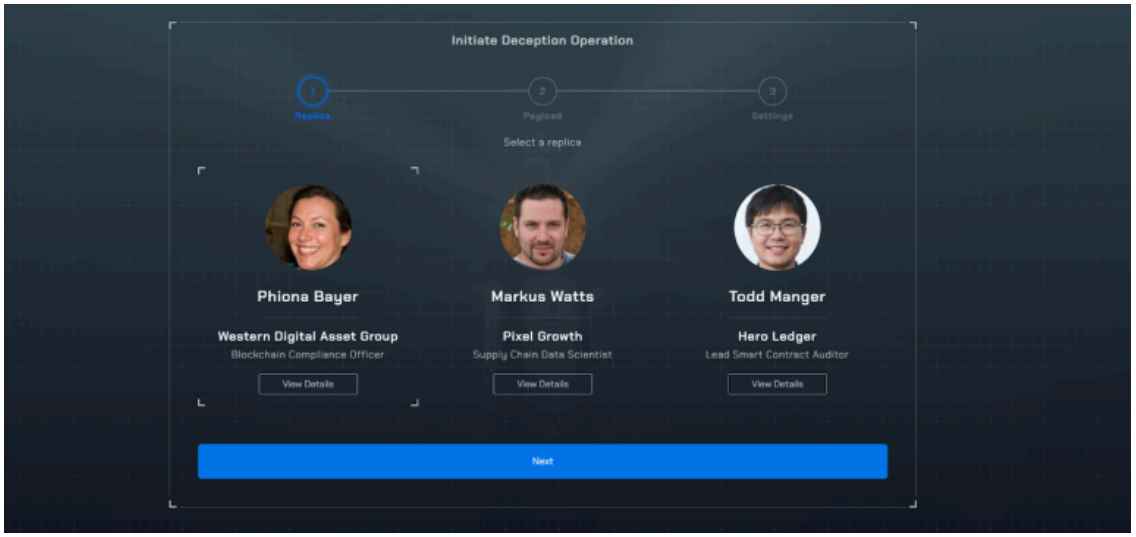
it's running in a legitimate corporate environment. Long-term observation can be beneficial for a couple reasons, most notably:

- Advanced malware often "sleeps" for long periods, waiting for an elapsed period of time before continuing execution or downloading additional payloads.
- When the analyst wants to observe additional payload drops (for example, in a loader scenario) or hopes to catch hands-on-keyboard actions or follow-up objectives the attackers are trying to execute.

Pretend for a moment I'm a malware analyst (which I am, so there's not much stretch of the imagination here). I detonated an unknown malware sample in my own virtual machines as well as several commercial sandboxes. Using publicly available commercial and free sandboxes, I determined that the malware belongs to a popular loader family. *(Loaders are a class of malware that download additional payloads. They typically perform sandbox detection and other evasion techniques to ensure the target system is "clean" before executing the payload.)*

I know this malware is a loader, but I want to understand what payload it ultimately drops. This behavior isn't observable in the other sandboxes I've tried. I suspect that's because the malware only communicates with its C2 and deploys its payload after a long period of time. I then submit the malware sample to Deception Pro.

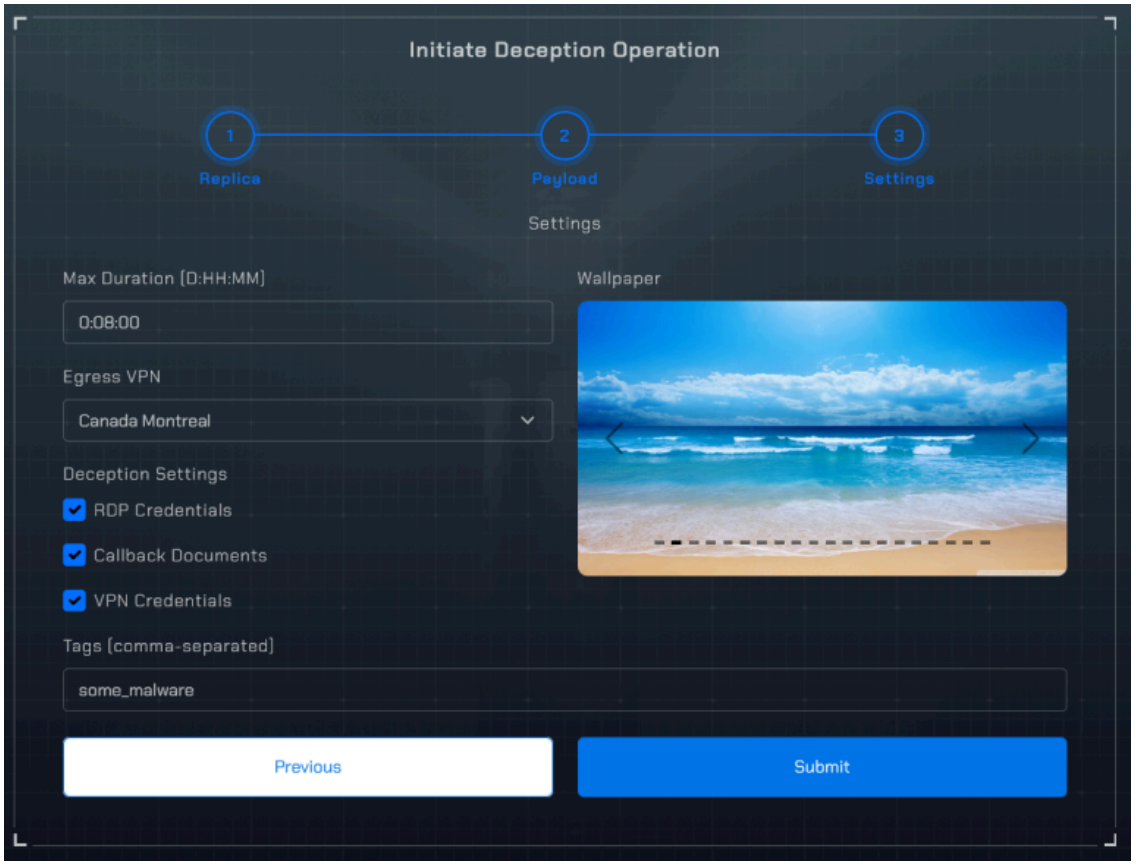
When starting a new Deception Pro session, you're greeted by an "Initiate Deception Operation" menu, which is a cool, spy-like way of saying, "start a new sandbox run." James Bond would approve.



In this menu, we can choose from one of three randomly generated profiles, or "replicas," for the user account in your sandbox – essentially, your "target." This person works for one of the randomly generated company names and is even assigned a fancy title. Deception Pro then generates fake data to populate the sandbox environment, and this replica acts as a starting point or seed. I chose Mr. Markus Watts, a Supply Chain Data Scientist at the company Pixel Growth. Looks legit to me.

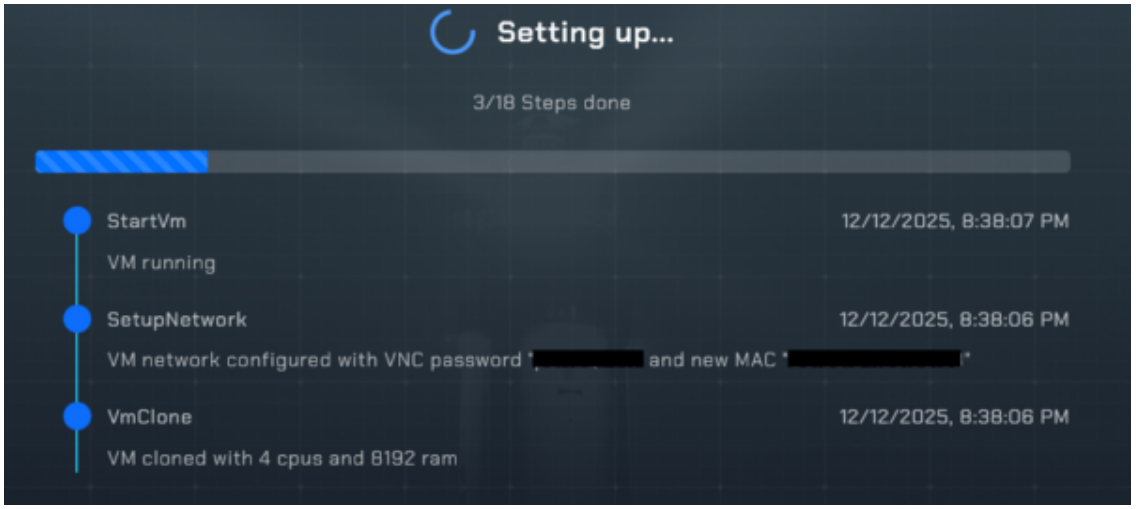
In the next menu, we're prompted to upload our malware sample and choose additional details about the runtime environment. The two primary options are **"Detonate Payload"** and **"Stage Environment Only."** Detonate Payload does what you'd expect and immediately detonates the payload once the environment spins up. Stage Environment Only allows the operator (you) to manually interact with the analysis environment. I haven't

experimented with this option.The final menu before the sandbox starts is the **Settings** menu. Here, we can select the detonation runtime (days, hours, minutes), the egress VPN country, some additional settings, and most importantly, the desktop wallpaper of the user environment. I'll choose a relaxing beach wallpaper for Mr. Watts. He probably needs a nice beach vacation after all the work he does at Pixel Growth.



As Deception Pro is designed for long-term observation, it's best to set a longer duration for the run. Typically, I set it to 5–8 hours, depending on my goals, and I've had good results with this.

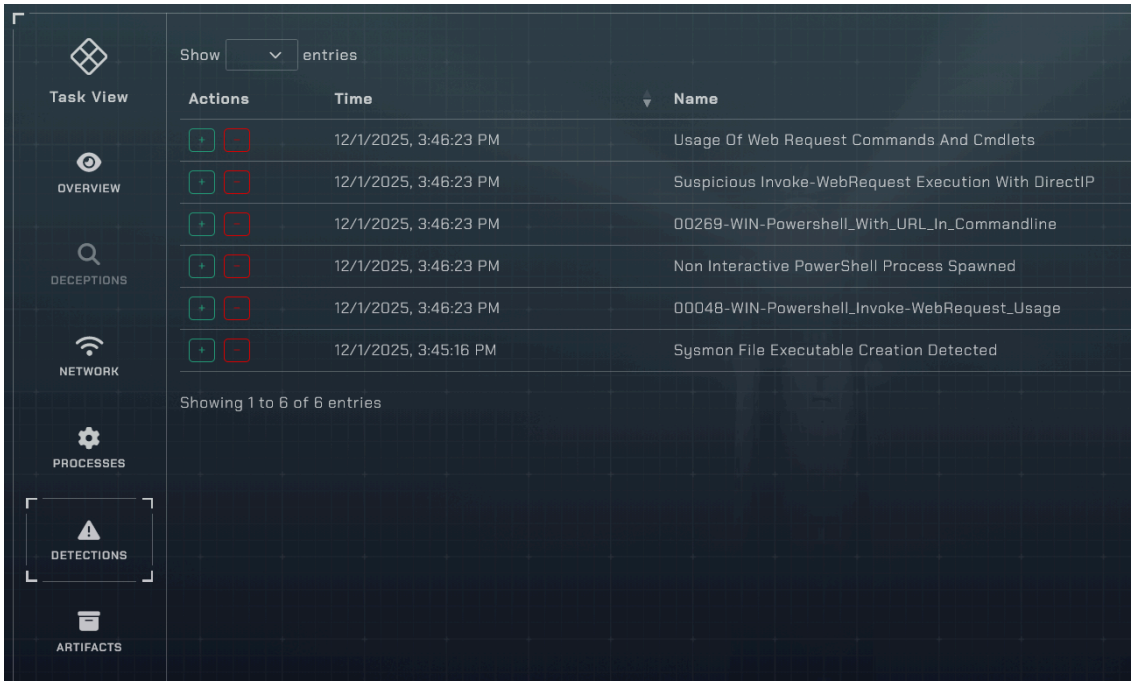
After clicking the **Submit** button, the analysis environment is set up and populated with random dummy data, such as fake files, documents, and other artifacts, as well as an entire fake domain network. This creates a realistic and believable environment for the malware to detonate in.



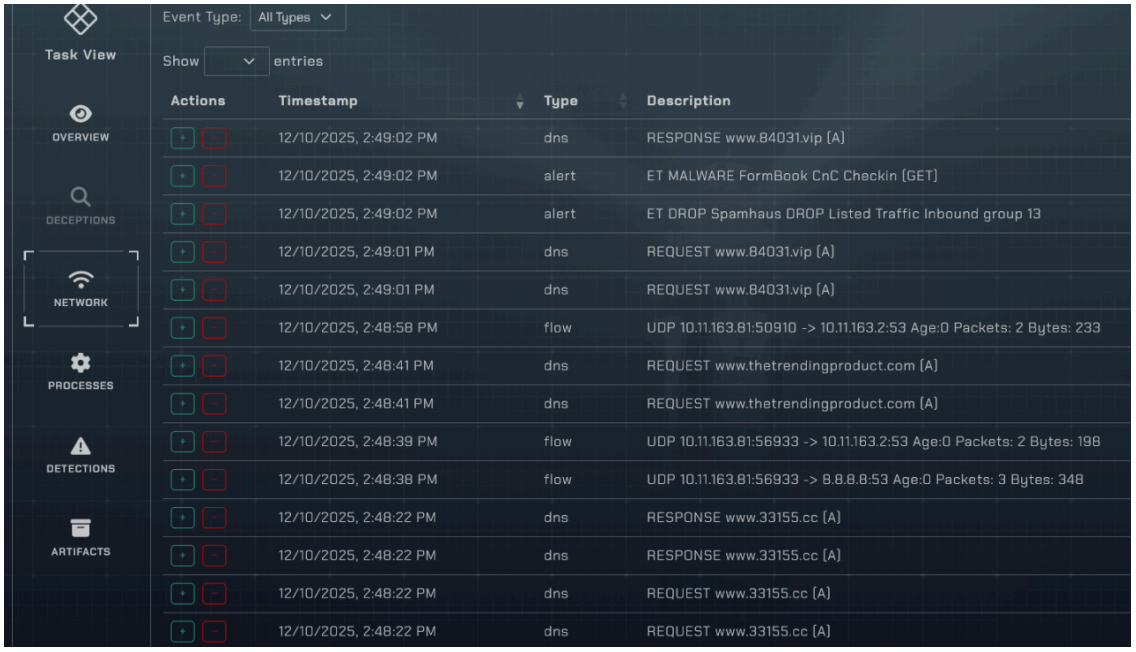
Behavioral and Network Analysis

Fast-forward eight hours, and our analysis is complete. I'm excited to see what behaviors were captured. We'll start with the **Reports → Detections** menu.

The Detections menu shows key events that occurred during malware detonation. There are a few interesting entries here, including suspicious usage of Invoke-WebRequest and other PowerShell activity. Clicking on these events provides additional details:

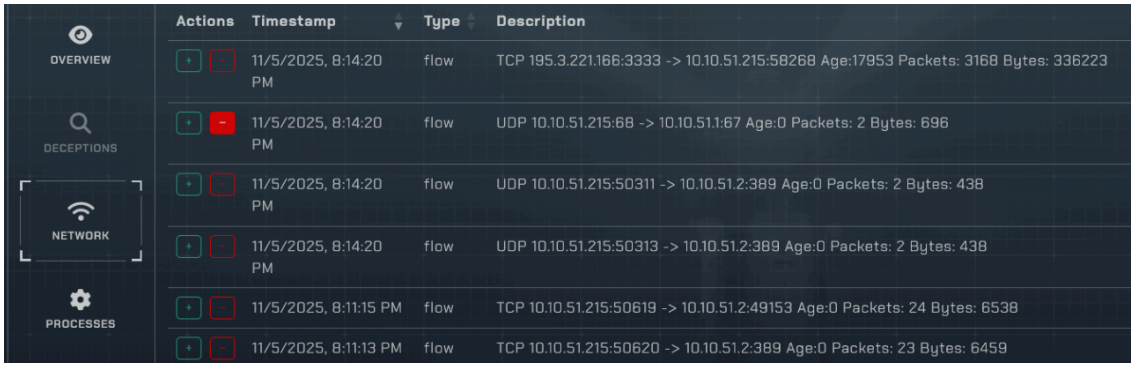


In the **Network** tab, we can view network connections such as HTTP and DNS traffic, along with related alerts:



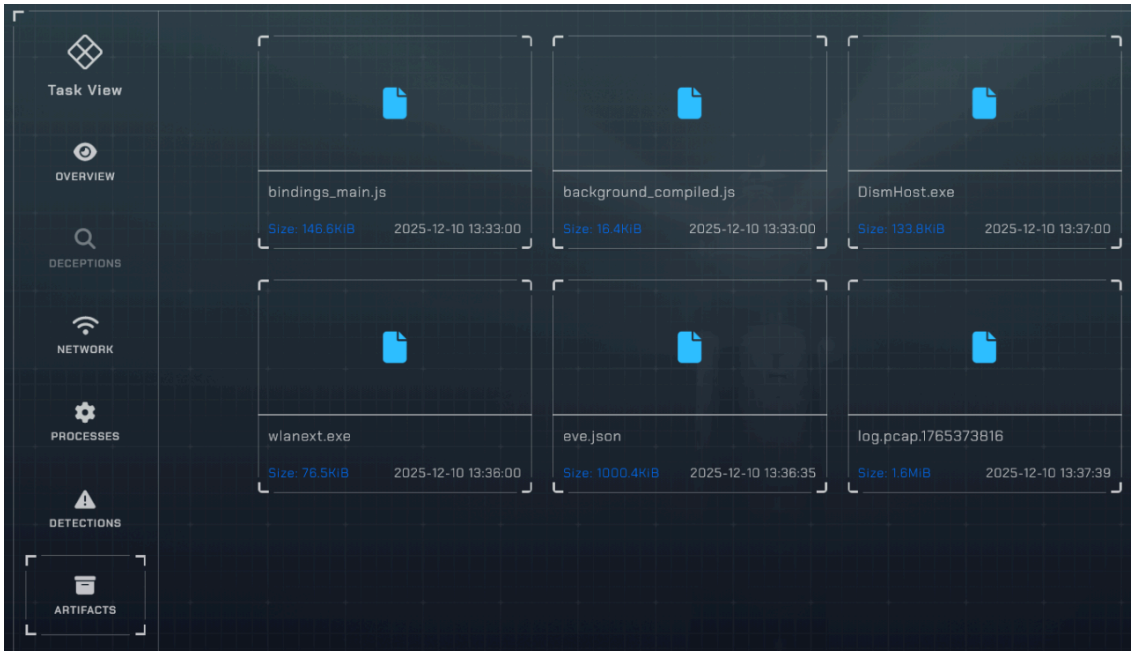
In the screenshot above, you may notice several web requests as well as a network traffic alert for a “FormBook C2 Check-in.” This run was indeed a FormBook sample, and I was able to capture eight hours of FormBook traffic during this specific run.

I was also able to capture payload downloads in another run:



In this run (which was a loader), a 336 KB payload was delivered roughly five hours into execution. This highlights the fact that some loaders delay payload delivery for long periods of time.

The **Artifacts** menu allows analysts to download artifacts from the analysis, such as PCAPs, dropped files, and additional downloaded payloads:



Regarding PCAPs, there is currently no TLS decryption available, which is a drawback, so let’s touch on this now.

Conclusions

It’s important to remember that Deception Pro is a specialized sandbox. I don’t believe it needs to have all the features of a traditional malware sandbox, as that could cause it to become too generalized and lose its primary strength: creating believable target users and lightweight environments while enabling long-term observation of

malware and follow-on actions. Here are some of the benefits I noticed when using Deception Pro, and some potential room for improvement:

Benefits

- Generates operating environments that simulate very realistic enterprise networks. This can expose additional malware and threat actor activities that other sandboxes may miss, like pivoting or network reconnaissance.
- Allows users to specify long detonation runtimes (hours to days) for observance of full attack chains (from initial infection to command and control, data exfiltation, and additional module and payload drops.
- Captures key events, behaviors, and network traffic of interest for investigators and researchers
- Allows interaction with the running sample and environment

Room for Improvement

- PCAP decryption is currently missing (though this is reportedly coming)
- Behavioural output is somewhat limited in its current state. This wasn't too detrimental for my use case, as I primarily used Deception Pro as a long-term detonation environment rather than a full-fledged analysis sandbox. I rely on other tools and sandboxes for deeper analysis.
- Currently no memory dump capabilities or configuration extraction

Also, note that the operating system environment is randomly generated, which limits customization (such as usernames, company names, etc.). This will rarely be an issue, but could matter when attempting to detonating highly targeted malware.

Overall though, I think the team behind Deception Pro is well on its way to creating a solid specialty sandbox, and I'm excited to see where it goes. Big thanks to Paul and the team for letting me spam their servers with malware.
