Covenant v0.5

Ryan Cobb · Follow
Published in Posts By SpecterOps Team Members · 5 min read · Jun 4, 2020

Intro

Covenant v0.5 is being released today and includes a set of new features, a major UI update, and lots of bug fixes.

This will be a short post documenting some of the major feature updates.

Blazor: Front-End C#

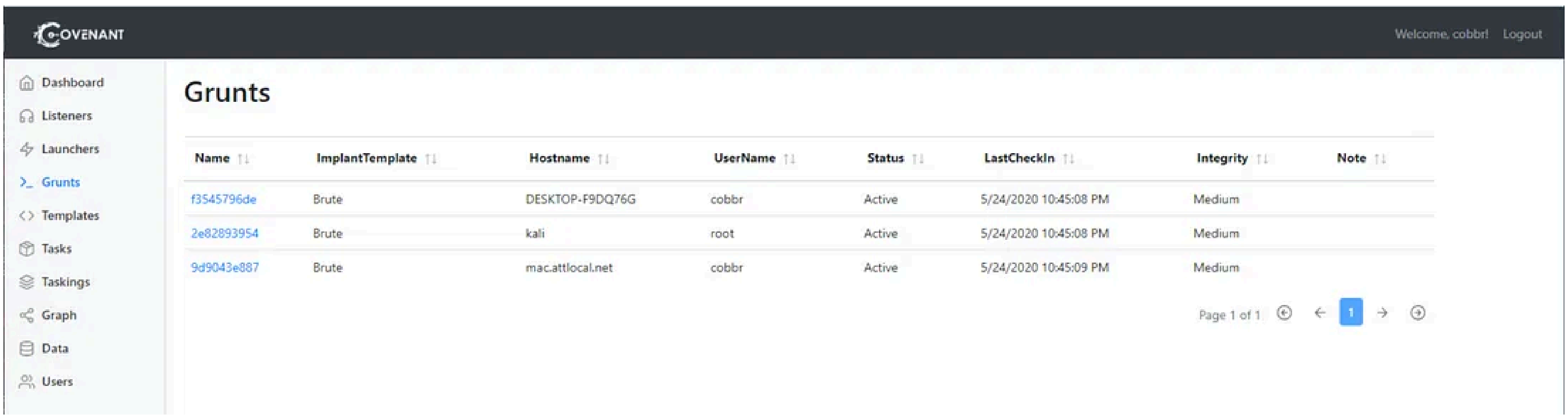
The largest update in v0.5 is migrating the interface over to the Blazor framework. Blazor is a framework for writing client-side web interfaces in you visit the Covenant site an HTTP request is made to pull down the initial ap

in

The main difference you'll see as a user, is pages should load a little bit faster than before, and the entire interface should update in real-time as things change on the backend server. For example, no more need to refresh the page to check for a Grunt's new LastCheckIn time! The interface has been overhauled to take advantage of the real-time features provided by Blazor.

Brutes: The New .NET Core Implant

Covenant has a brand new built-in, cross-platform implant: Brutes! Operators are no longer limited to Grunt implants or to Windows targets.



Grunts

Name	ImplantTemplate	Hostname	UserName	Status	LastCheckIn	Integrity	Note
f3545796de	Brute	DESKTOP-F9DQ76G	cobbr	Active	5/24/2020 10:45:08 PM	Medium	
2e82893954	Brute	kali	root	Active	5/24/2020 10:45:08 PM	Medium	
9d9043e887	Brute	mac.attlocal.net	cobbr	Active	5/24/2020 10:45:09 PM	Medium	

Active Brutes on Windows, Linux, and MacOS Targets

Covenant has slowly been moving towards the goal of being truly implant-agnostic, and the introduction of the Brute implant is a major step towards

The Brute implant itself is fairly simple, there isn't a huge library of available tas

ps e percentage of it's codebase with Grunts, with some minor changes to port the code over to .NET Core, but there's not many libraries to borrow tasks from available in .NET Core like there is for the .NET Framework, such as SharpSploit and GhostPack.

There's a lot to say about the potential of .NET Core implants, and I plan to release a follow-up post on .NET Core malware in the near future. I'll leave most of those details for that post. For now, it's enough to say that Brutes are cross-platform .NET Core implants with a limited set of base functionality.

Shareable Tasks

Tasks can now be easily imported and exported to or from a running Covenant instance as YAML files. This makes it much easier to share your Covenant tasks with others that may be interested in your work.

It also makes it much easier to contribute built-in Tasks to the project. Before this change, built-in Tasks were described programmatically in Covenant's source code, and required source code changes to add new built-in Tasks. Now, YAML files included under the `Data\Tasks` directory will automatically be parsed and imported when Covenant first starts up. Additional tasks that you would like to use can be imported within the interface.

Other Minor Updates

Th

mention here, but here's a few other minor updates you'll find in v0.5.

- **Author Information** — Author information is included in GruntTasks, so you'll get credit for the Tasks you contribute to Covenant.
- **Hide and Unhide Grunts** — The ability to hide and unhide Grunts has been reworked, so you can easily hide and unhide old Grunts that are no longer in use from the UI.
- **Manual Data Creation** — Credentials and Indicators can be manually added to the schema now, instead of just relying on Covenant's automatic detection of new data.
- **Updated ReferenceSourceLibraries** — ReferenceSourceLibraries such as SharpSploit and GhostPack have been brought up to date with the public versions.

Conclusion

Hopefully this has given you a brief summary of some of the major updates in the new v0.5 Covenant update. This has been the first major Covenant update in nearly 7 months, as the UI conversion over to Blazor took much more work than anticipated. Hopefully it's been worth the wait!

In the future, we hope that Covenant updates will be much more frequent and incremental, to decrease the waiting time between updates. In the near term future, Covenant has developed quite a backlog of issues and pull requests that need to be addressed, and we likely will be focused on those

223 Followers · Editor for Posts By SpecterOps Team Members

More from Ryan Cobb and Posts By SpecterOps Team Members

Ryan Cobb in Posts By SpecterOps Team Members

Designing Peer-To-Peer Command and Control

In this post we will discuss the design and implementation of peer-to-peer command…

May 1, 2019 81 1

Will Schroe… in Posts By SpecterOps Team Mem…

Certified Pre-Owned

Active Directory Certificate Services has a lot of attack potential!

Jun 17, 2021 489 4

Hope Walk… in Posts By SpecterOps Team Memb…

An Introduction to Manual Active Directory Querying with Dsquery…

Introduction

Ryan Cobb in Posts By SpecterOps Team Members

Operational Challenges in Offensive C#

As offensive toolsets continue to move

Satyam Pathania in InfoSec Write-ups

Why I Don’t Recommend People To Get into Cybersecurity?

Cybersecurity isn’t always what it seems — it’s tough, demanding, and stressful.

✦ Oct 24 388 6

Aardvark Infinity in Aardvark Infinity

Dragon

Description: A formidable C# tool designed for advanced Active Directory (AD)...

✦ Sep 28

Lists

Medium's Huge List of Publications Accepting...

378 stories · 3815 saves

Forrest Kas... in Posts By SpecterOps Team Mem...

RED TEAM

Sep

See more recommendations