SPECTEROPS

MYTHIC

Mythic 2.3 & Apollo 2.0

February 23, 2022

Who Are We?

Cody Thomas
@its_a_feature_
Mythic Author
Engineer

Dwight Hohnstein
@djhohnstein
Apollo Author
Senior Consultant

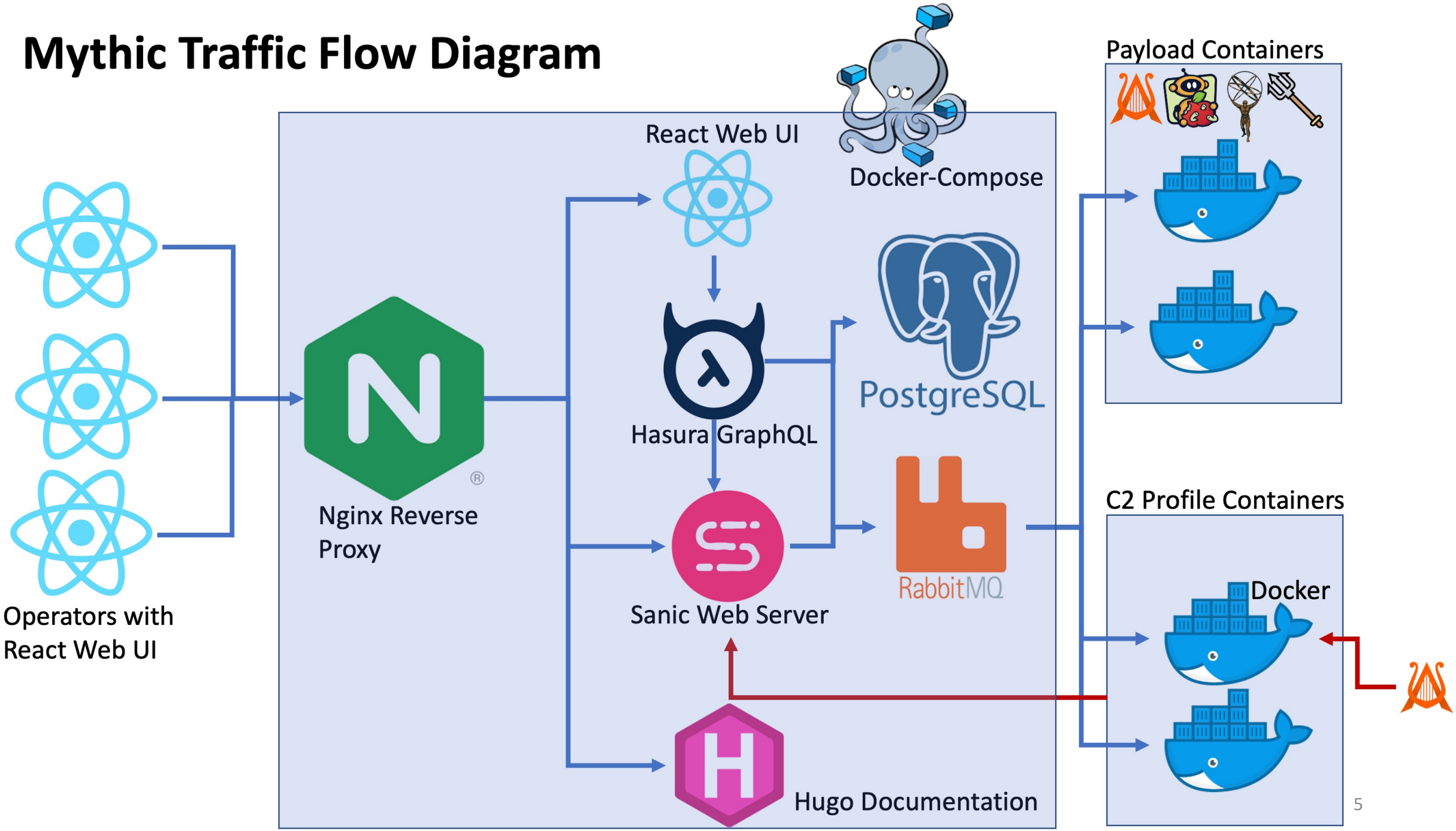
Mythic 2.3 & Apollo 2.0 Webinar Overview

- What is Mythic?
- What is Apollo?
- What's new in Mythic 2.3?
  - A case study in Apollo 2.0
- What's coming next?
- Q&A

What is Mythic?

- Open-Source Red Teaming Framework
  - https://github.com/its-a-feature/Mythic
  - https://docs.mythic-c2.net/
- Modular Framework using Docker
- Multi-Operator, Muti-Operation, Web-based
  - Access Controls, Spectator Mode
  - IoC/Artifact Tracking
- Plug-n-Play architecture for Agents and C2 Profiles
  - https://github.com/MythicAgents
  - https://github.com/MythicC2Profiles
- MITRE ATT&CK Mapping and Tracking

Mythic Traffic Flow Diagram



What is Apollo?

- Open-Source C# Windows Agent for Mythic
  - https://github.com/MythicAgents/apollo
- Supports many Mythic features:
  - SOCKS5, File Browser, Subtasking, Credentials, and more
- Provides many OPSEC features:
  - PPID spoofing, customized injection, sacrificial processes, etc

Mythic Demo – Basic Usage

1. Starting Mythic
2. C2 Profiles / Payload Types
3. Creating Payloads
4. Getting a Callback
5. Issuing commands

https://www.youtube.com/watch?v=wprAyoQys-o&list=PLHVFedjbv6sOnsGlxslDiGFqsiY7eARJD&index=1

Mythic Demo Recap

- Git Clone Mythic
  - git clone https://github.com/its-a-feature/Mythic
- Edit any configurations you want in a .env file
- Start Mythic
  - sudo ./mythic-cli start
- View exposed ports and services in status output
  - sudo ./mythic-cli status
- Create poseidon payload

Tasking Agents Demo – CLI and Modals

1. Help
2. Loaded Commands
3. Tab Complete
4. Popup Modals

https://www.youtube.com/watch?v=hGqIc2-XpJE&list=PLHVFedjbv6sOnsGlxslDiGFqsiY7eARJD&index=2

Tasking Agents Recap

- Click "Interact" to start tasking an agent
- Tab-complete loaded commands
- Tab-complete parameters
- Parsed CLI into named parameters
- Tasking modal
  - Shift+Enter always opens the modal

Parameter Groups – What Are They?

- Commands are multi-purpose with distinct sets of parameters
  - Inspired by PowerShell's Parameter Sets
- Parameter Groups resolve ambiguity between desired actions
  - sc_create, sc_query, sc_start, etc. vs sc
- Modals useful in determining required and optional parameters
- Parameter Groups are inferred based on supplied arguments
- Parameter Groups in the Modal popup are explicitly selected

Parameter Group Demo – SC

1

Popup Modal

2

Select Parameter Group

3

Tab Complete Arguments

https://www.youtube.com/watch?v=vVcJ7ApH87E&list=PLHVFedjbv6sOnsGlxslDiGFqsiY7eARJD&index=3

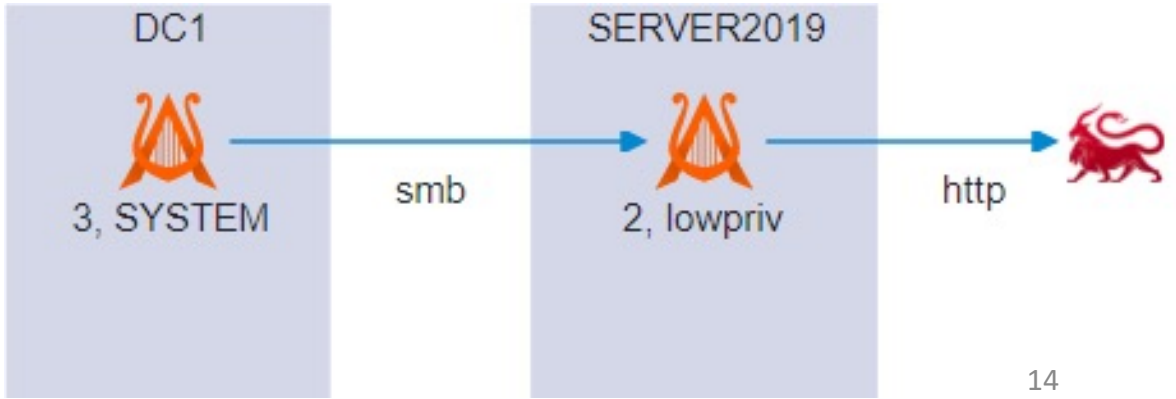
P2P Demo – PSEXEC

1. Create Apollo Shellcode
2. Generate a Service Executable
3. Upload Service Executable to Remote Host
4. Create and Start Service
5. Link and Issue P2P Tasking

https://www.youtube.com/watch?v=wewlWejnpCQ&list=PLHVFedjbv6sOnsGlxslDiGFqsiY7eARJD&index=4

P2P Demo Recap

- Generate a Payload for Lateral Movement
- Use Token Manipulation to Access DC1 via "make_token"
- Upload Service Executable using "upload"
- Create and Start ApolloSvc via "sc"
- Link to DC1 using Named Pipes



SOCKS5 Demo

1

Start SOCKS5 Proxy

2

Connect Proxychains4

3

RDP to Internal Computer

4

Proxy Tools like Impacket

https://youtu.be/XoSzkPYsyDE

MITRE ATT&CK Demo

1. Command Mappings
2. Task Mappings
3. Payload Mappings
4. Export to Navigator Layer

https://www.youtube.com/watch?v=9WxsK4zdp3g&list=PLHVFedjbv6sOnsGlxslDiGFqsiY7eARJD&index=5

MITRE ATT&CK Demo Recap

- All commands can map to MITRE ATT&CK
  - We can view the "realm of the possible" by viewing all these mappings at once
  - We can also view this mapping by specific Payload Types
- If a command is mapped to ATT&CK, then Tasks based on it are also mapped
  - We can view this "realm of the actual" to see what all we've done so far in an operation
- We can also tag issued tasks and filter our mapping by just a single tag
- All displays can be exported to ATT&CK Navigator Layers

Mythic – Going Forward

- Mythic Scripting Updated for GraphQL
  - Currently hits the old REST / WebSocket interfaces
- Mythic Developer Series
  - Blog/YouTube Series
  - Based on feedback from survey
    - https://www.surveymonkey.com/r/MythicDeveloperSeries
- Updating the back-end Mythic Server
  - Split out RabbitMQ, Authentication, Agent Message Parsing
  - Move to Golang instead of Python3
  - Provide more language support for agent / c2 containers
- Add more pages / features to the new UI
  - Show how people can extend the UI with their own custom React code

Apollo – Going Forward

- Unit Testing
  - Ensures continuity of functionality across update cycles
- C2 Profiles
  - DynamicHTTP
  - Third-Party (Slack, OWA, etc.)
  - Modular, Rotating, Profiles
- Evasion Options
  - Currently Only Injection Techniques
  - Rearchitect Commands for Compile Time Obfuscation

Q&A

- GitHub
  - https://github.com/its-a-feature/Mythic
  - https://github.com/MythicAgents/apollo
- Documentation
  - https://docs.mythic-c2.net
- #Mythic channel in BloodHound Slack
  - https://bloodhoundgang.herokuapp.com/