Agoric Vaults Implementation Assessment

Security Assessment Report

Agoric

Prepared for Agoric Systems
Operating Company
April 28, 2023 (version 1.0)

Project Team:

Technical Testing	Brandon Perry and Joshua Domangue
Technical Editing	Nathan Keltner and Sara Bettes
Project Management	Sara Bettes

Table of Contents

Engagement Overview

Assessment Components and Objectives

Agoric Systems Operating Company ("Agoric") recently engaged Atredis Partners ("Atredis") to perform a Vaults Implementation Assessment of the Agoric platform. Objectives included validation that Agoric infrastructure and services were developed and deployed with security best practices in mind and to obtain third party validation that any significant vulnerabilities present in Agoric's environment were identified for remediation.

Testing was performed from April 3 through April 21, 2023, by Brandon Perry and Joshua Domangue of the Atredis Partners team, with Sara Bettes providing project management and delivery oversight. For Atredis Partners' assessment methodology, please see Appendix I of this document, and for team biographies, please see Appendix II. Specific testing components and testing tasks are included below.

COMPONENT	ENGAGEMENT TASKS
Agoric Vaults Implementation Assessment	
	• Assess vault initialization process • Assess handling of vault-related governance parameters • Assess whether IST remains over-collateralized through volatile market cycles • Assess whether attackers can force, delay, or otherwise tamper with liquidity events • Assess whether attackers can cause unexpected debt or shortfall • Assess whether attackers can affect protocol earnings • Assess whether attackers can affect outputs from the pricing oracle network
Reporting and Analysis	
Analysis and Deliverables	• Status Reporting and Realtime Communication • Comprehensive Engagement Deliverable • Engagement Outbrief and Remediation Review

The ultimate goal of the assessment was to provide a clear picture of risks, vulnerabilities, and exposures as they relate to accepted security best practices, such as those created by the National Institute of Standards and Technology (NIST), Open Web Application Security Project (OWASP), or the Center for Internet Security (CIS). Augmenting these, Atredis Partners also draws on its extensive experience in secure development and in testing high-criticality applications and advanced exploitation.

Engagement Tasks

Atredis Partners performed the following tasks, at a high level, for in-scope targets during the engagement.

Application Penetration Testing

For relevant web applications, APIs, and web services, Atredis performed automated and manual application penetration testing of these components, applying generally accepted testing best practices as derived from OWASP and the Web Application Security Consortium (WASC).

Testing was performed from the perspective of an anonymous intruder, identifying scenarios from the perspective of an opportunistic, Internet-based threat actor with no knowledge of the environment, as well as, from the perspective a user working to laterally move through the environment to bypass security restrictions and user access levels.

Where relevant, Atredis Partners utilized both automated fuzzing and fault injection frameworks as well as purpose-built, task-specific testing tools tailored to the application and platforms under review.

Binary and Runtime Analysis

For relevant software targets identified during the course of this engagement, Atredis performed binary and runtime analysis, using debugging and decompilation tools to analyze application flow to aid in software security analysis. Where relevant, purpose-built tools such as fuzzers and customized network clients may have been utilized to aid in vulnerability identification.

Configuration and Architecture Review

Atredis Partners performed a high-level review of available documentation and configuration data with an eye toward the overall functional design and soundness of the implementation. A key aspect of this component was identifying gaps in the architecture and design regarding aspects of design that reduce overall defensibility, aimed at pointing out fundamental issues in the application architecture that should be addressed early in the development cycle as opposed to later when the platform is closer to a full production state.

While specific vulnerabilities may be identified during the architecture and configuration review, the intent is less on finding individual defects and more on how the design of a given target affects its overall defensibility. Outcomes of the architecture review helped inform testing objectives throughout the rest of the engagement while also helping the client define a long-term platform maturity and security design roadmap.

Network Protocol Analysis

With the objective of identifying scenarios where the integrity of trusted communications can be diminished or reduced, Atredis Partners reviewed network traffic using various packet flow analysis and packet capture tools to observe in-scope network traffic. Network communications were analyzed for the presence of cleartext communications or scenarios where the integrity of cryptographic communications can be diminished, and Atredis attempted to identify means to bypass or circumvent network authentication or replay communications, as well as other case-dependent means to abuse the environment to disrupt, intercept, or otherwise negatively affect in-scope targets and communications.

Source Code Analysis

Atredis reviewed the in-scope application source code, with an eye for security-relevant software defects. To aid in vulnerability discovery, application components were mapped out and modeled until a thorough understanding of execution flow, code paths, and application design and architecture is obtained. To aid in this process, the assessment team engaged key stakeholders and members of the development team where possible to provide structured walkthroughs and interviews, helping the team rapidly gain an understanding of the application’s design and development lifecycle.

Executive Summary

To perform testing of the new Vaults implementation, Atredis was granted access to three governance keys for the `Xnet` network used for testing by Agoric. This testing environment could be redeployed by Agoric with a clean slate or updated smart contract features throughout the engagement if needed. Atredis also created their own user keys with the Keplr smart wallet extension for Chrome.

Atredis used a single-page web application deployed on IPFS to view and manage vaults on the `Xnet` chain for governance entities as well as our own Keplr users. Finally, Atredis used the governance application deployed at https://econ-gov.inter.trade/ for vault parameter governance and the smart wallet dashboard at https://main.wallet-app.pages.dev/wallet/ to facilitate actions related to vault functionality.

Key Conclusions

Atredis found one low-severity issue related to the vault initialization process. Governance parameters could only be modified by officially invited members. The values that could be set for governance were determined by ratios using numbers that could not be negative. Similarly, oracles could only be invited via officially signed invitations and Atredis found no way to forge oracle invitations. Even if an attacker could trick someone into creating an official invitation, the ability to affect the prices reported by the oracle middleware would still be hampered by Agoric's method of price selection.

Atredis found no ability to mint `IST` without having appropriate collateral and found no way to keep minted `IST` when collateral became insufficient, except in extreme circumstances noted in this finding. Atredis tested the oracle middleware for unauthorized on-chain messaging, as well as the ability of an oracle to disrupt the general network or operations for users. Atredis could not publish oracle messages without official invitations and could not forge official invitations.

As in any security assessment, some general areas for improvement were noted, but overall Atredis Partners would rate Agoric's platform as sound from a security perspective and well-aligned with modern secure web application development practices.

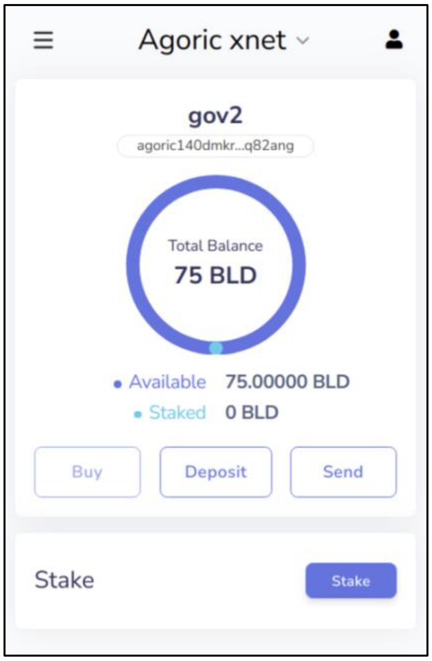
Platform Overview

Agoric's Vaults implementation is built on the Zoe framework and a custom JavaScript runtime. The current implementation relies on Agoric's chain networks for consuming `IST` and `ATOM` assets. Several security features protect smart contracts written in combination with the Zoe framework and the custom JavaScript implementation. Some of these features are an Object Capability (OCAP)-based security model and usage of natural numbers with no wraparounds, overflows, or underflows possible.

Specific to the scope of this assessment, the Vaults implementation relies on several pieces to create an end-to-end workflow of vault creation and liquidation.

Inter UI Distributed Application

The Inter UI Distributed Application (DApp) enables users to create and manage Vaults within the Agoric ecosystem. Authentication and authorization rely on the Keplr cryptocurrency wallet extension for Chrome but could use other wallets in the future. Regarding authorization, requests were approved through the Vaults DApp and the Keplr wallet authorization prompts.



Agoric xnet

gov2

agoric140dmkr...q82ang

Total Balance
75 BLD

Available 75.00000 BLD
Staked 0 BLD

Stake

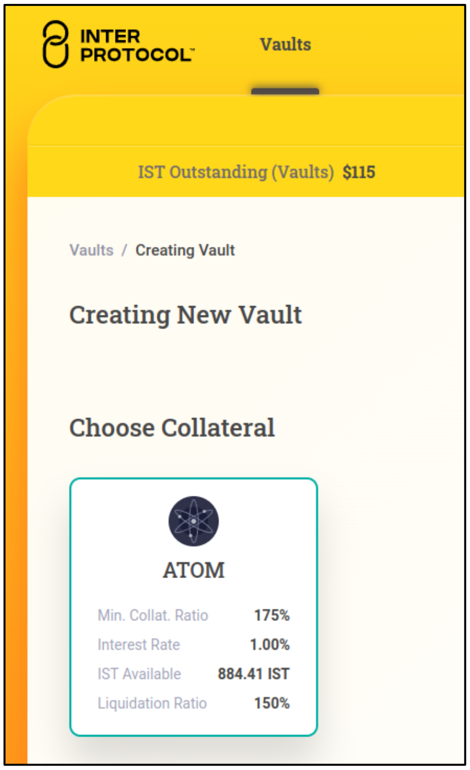
Keplr Smart Wallet on Agoric Test Network

The Inter UI DApp is currently stored on IPFS and is accessed via a web browser from the following URL.

```
https://bafybeidkbriy5kvmsr24nef5m3omtiztrrma2cjafryaqlh5byrrqdisw4.ipfs.cf-
ipfs.com/?wallet=main#/vaults
```

Vaults DApp URL

The application runs as a client-side single-page JavaScript application. Once authenticated, a user can begin creating and viewing vaults with collateralized assets.



Min. Collat. Ratio	175%
Interest Rate	1.00%
IST Available	884.41 IST
Liquidation Ratio	150%

Authenticated View of Vaults DApp on IPFS

If the user has been invited to the economic committee, altering the chain parameters can be performed on another web application, such as governed collateral and liquidation requirements. Much of the functionality available in Inter UI is also implemented in the `agoric` command line utility. Atredis generally focused on the browser application, but augmented testing with usage of the command line utility where applicable.

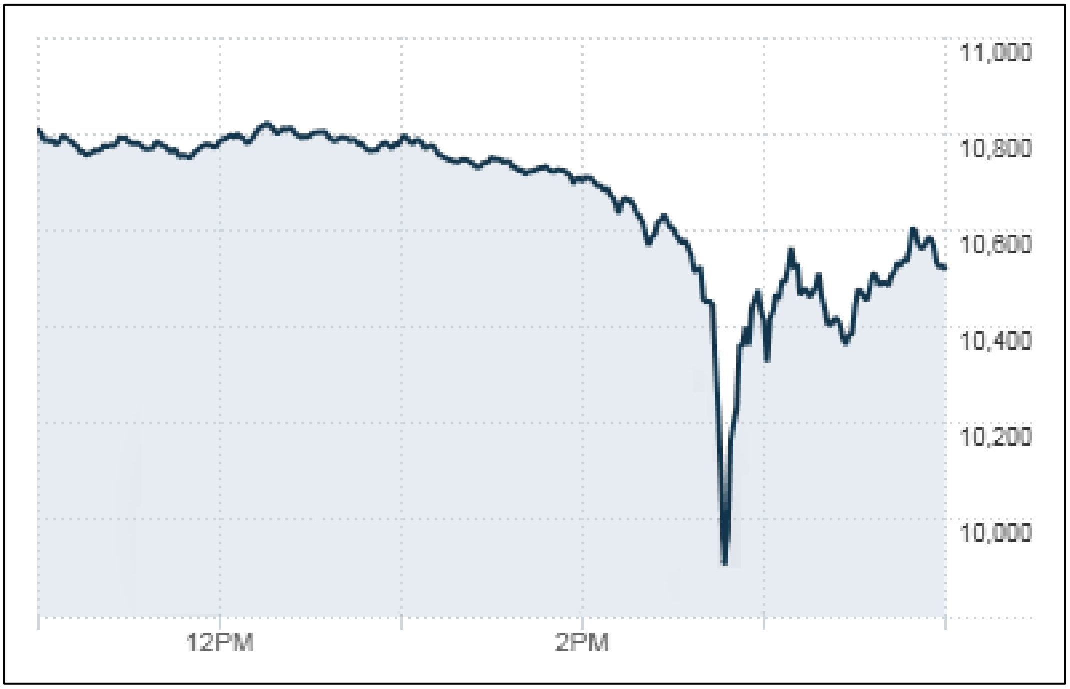
Oracle Middleware

The oracle middleware in the Agoric ecosystem allows smart contracts to retrieve specific asset prices in near real-time. The middleware is itself written on smart contracts in Agoric's JavaScript implementation. The middleware smart contracts allow inviting entities to publish on-chain messages regarding asset prices. Atredis found no way to publish on-chain oracle pricing messages without accepting an official invitation first.

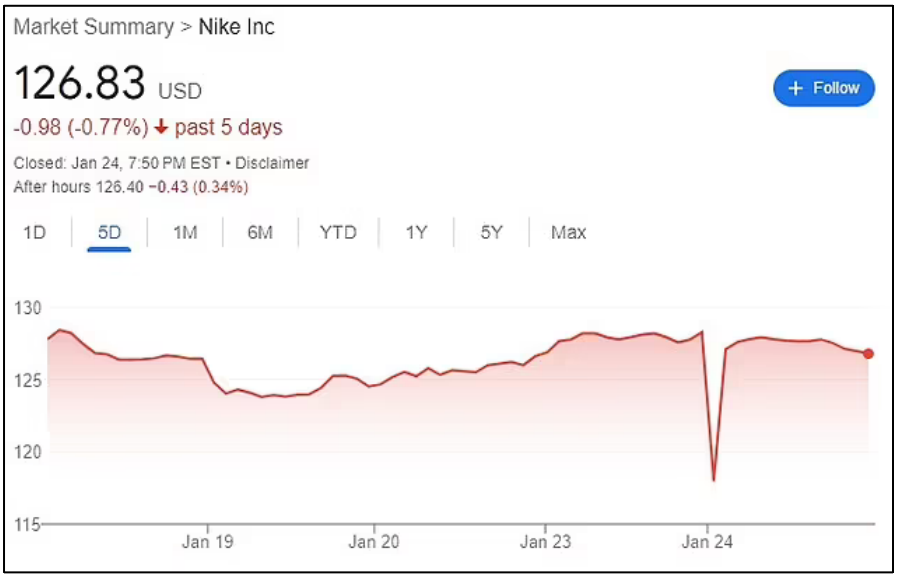
Prices are pushed via messages on to the chain for bidding agents to consume. For a price of an asset to be pushed and acted on by the oracle middleware, it must be greater than 0. An asset value reported by an oracle that is 0 will be ignored by the Agoric smart contracts. This could have unintended side-effects, as noted in this finding. Oracle agents can't push prices for assets without accepting an invitation to do so.

Oracle prices are checked every ten minutes if no new prices have been pushed. Each governance member with the ability to kick off a round of price updates cannot run more than one update in a row. Another governance member must request another update before the first member can request a new update. Multiple sources for prices can be configured per asset type, allowing for less reliance on a central source.

The median price from multiple sources is used in calculations. This is an important feature (as opposed to using the average) is it creates a barrier to “glitch crashes” which have been shown with history to negatively affect other effective oracles in the real world.



Dow-Jones Industrial Average on May 6, 2010


Market Summary > Nike Inc

126.83 USD

-0.98 (-0.77%) ↓ past 5 days

Closed: Jan 24, 7:50 PM EST • Disclaimer

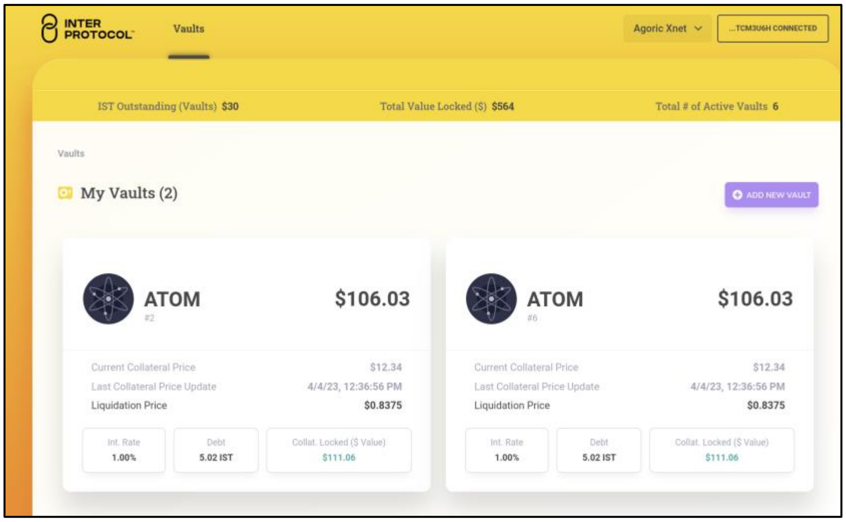
After hours 126.40 −0.43 (0.34%)

Nike Stock on Jan 24, 2023

When multiple sources are used and the median price is used instead of the average, a glitch or otherwise wildly untrue value should not negatively affect the median ***directly***. The range of values may shift. However, a glitched value in either direction would directly affect the average of multiple sources. Because the Agoric middleware determines an asset price via the median instead of the average of all prices, large swings due to a misreported price should not happen in the current oracle middleware implementation, providing some resiliency to glitched or wildly untrue oracles.

Agoric Vaults

Agoric Vaults allow users to create discrete purses for single asset types. Vaults are backed by collateral, and the current implementation mints and burns `IST` tokens for `ATOM` collateral.



IST Outstanding (Vaults) $30 Total Value Locked ($) $564 Total # of Active Vaults 6

Vaults

My Vaults (2)

ATOM #2 $106.03

Current Collateral Price $12.34
Last Collateral Price Update 4/4/23, 12:36:56 PM
Liquidation Price $0.8375

Int. Rate 1.00% Debt 5.02 IST Collat. Locked ($ Value) $111.06

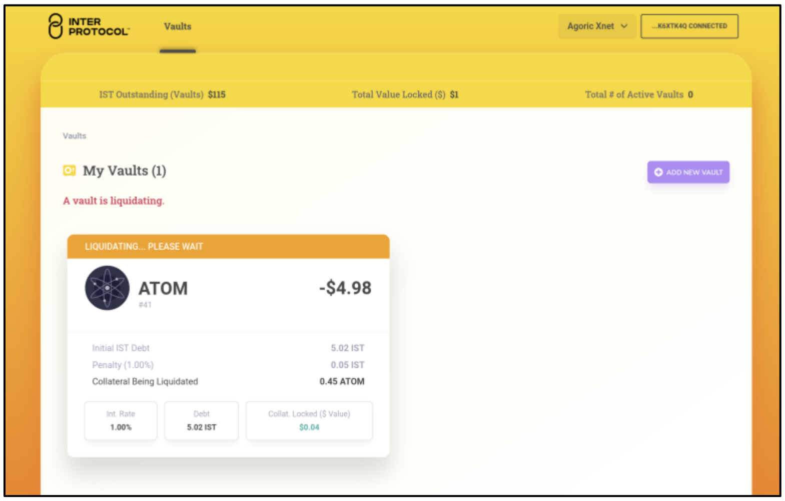
ATOM #6 $106.03

Current Collateral Price $12.34
Last Collateral Price Update 4/4/23, 12:36:56 PM
Liquidation Price $0.8375

Int. Rate 1.00% Debt 5.02 IST Collat. Locked ($ Value) $111.06

Vaults in Good Standing

Consuming the oracle agent on-chain messaging, the Vaults smart contract will liquidate collateralized vaults when the price of the collateral used to back the `IST` minted falls below a certain ratio. Vaults may be closed when in the active or liquidated state, but not when in a liquidation state. Atredis was unable to close a vault while in the liquidating state.



IST Outstanding (Vaults) $115 Total Value Locked ($) $1 Total # of Active Vaults 0

Vaults

My Vaults (1)

A vault is liquidating.

LIQUIDATING... PLEASE WAIT

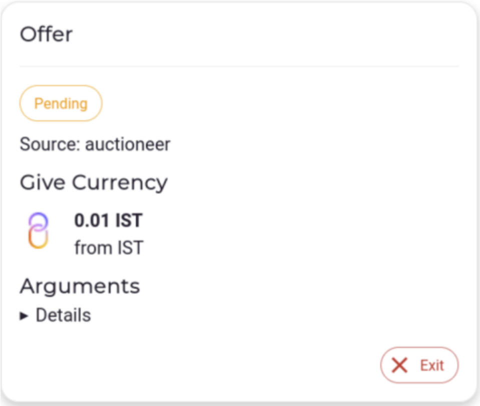
ATOM #41 -$4.98

Initial IST Debt 5.02 IST
Penalty (1.00%) 0.05 IST
Collateral Being Liquidated 0.45 ATOM

Int. Rate 1.00% Debt 5.02 IST Collat. Locked ($ Value) $0.04

Vault in Liquidating State

When vaults are liquidated, a modified descending clock auction is started where an auctioneer smart contract begins selling liquidated assets, dropping the price of the assets over time, and messaging the chain with the next asset price. Bidding agents consume these price messages and may bid on liquidated assets at discounted prices.



Offer

Pending

Source: auctioneer

Give Currency

0.01 IST
from IST

Arguments

▸ Details

Open Offer from Bidding Agent

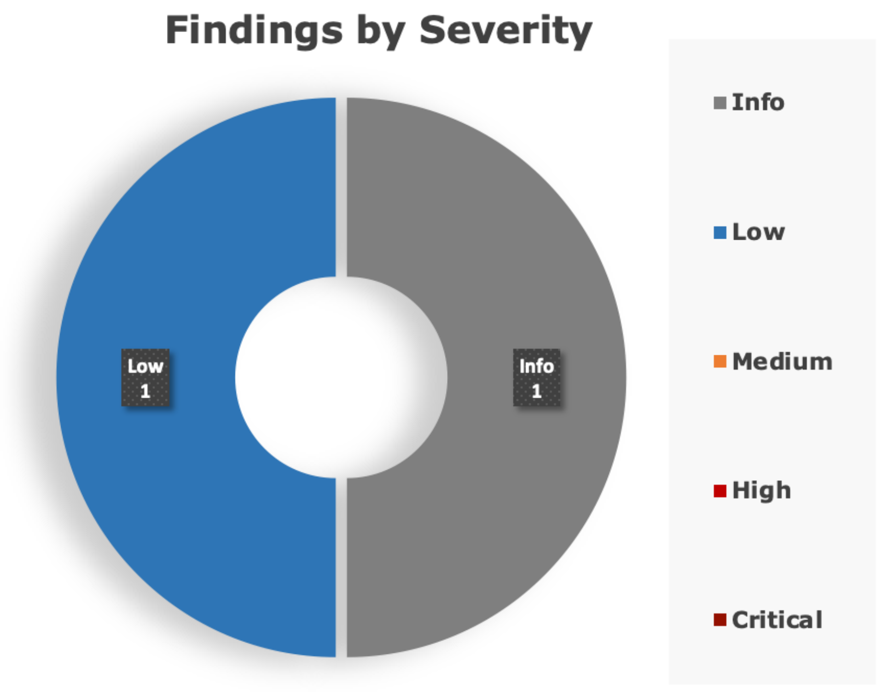
Various fees may be paid during the lifecycle of a vault. Notably, some fees are used to disincentivize under-collateralized vaults by charging a fee for forced liquidation. Fees are collected in the Reserve and are distributed pro rata.

Findings Summary

In performing testing for this assessment, Atredis Partners identified **one (1) low** severity finding and **one (1) informational** finding. No high or critical severity findings were noted. As stated earlier, none of these issues constitute a potential for direct compromise.

Atredis defines vulnerability severity ranking as follows:

- **Critical:** These vulnerabilities expose systems and applications to immediate threat of compromise by a dedicated or opportunistic attacker.
- **High:** These vulnerabilities entail greater effort for attackers to exploit and may result in successful network compromise within a relatively short time.
- **Medium:** These vulnerabilities may not lead to network compromise but could be leveraged by attackers to attack other systems or applications components or be chained together with multiple medium findings to constitute a successful compromise.
- **Low:** These vulnerabilities are largely concerned with improper disclosure of information and should be resolved. They may provide attackers with important information that could lead to additional attack vectors or lower the level of effort necessary to exploit a system.



Findings and Recommendations

The following section outlines findings identified via manual and automated testing over the course of this engagement. Where necessary, specific artifacts to validate or replicate issues are included, as well as Atredis Partners' views on finding severity and recommended remediation.

Findings Summary

The below tables summarize the number and severity of the unique issues identified throughout the engagement.

CRITICAL	HIGH	MEDIUM	LOW	INFO
0	0	0	1	1

Findings Detail

FINDING NAME	SEVERITY
Duplicate Offer ID Leads to Inaccessible Vaults	Low
Limitations of Automated Liquidation	Info

Duplicate Offer ID Creates Inaccessible Vaults

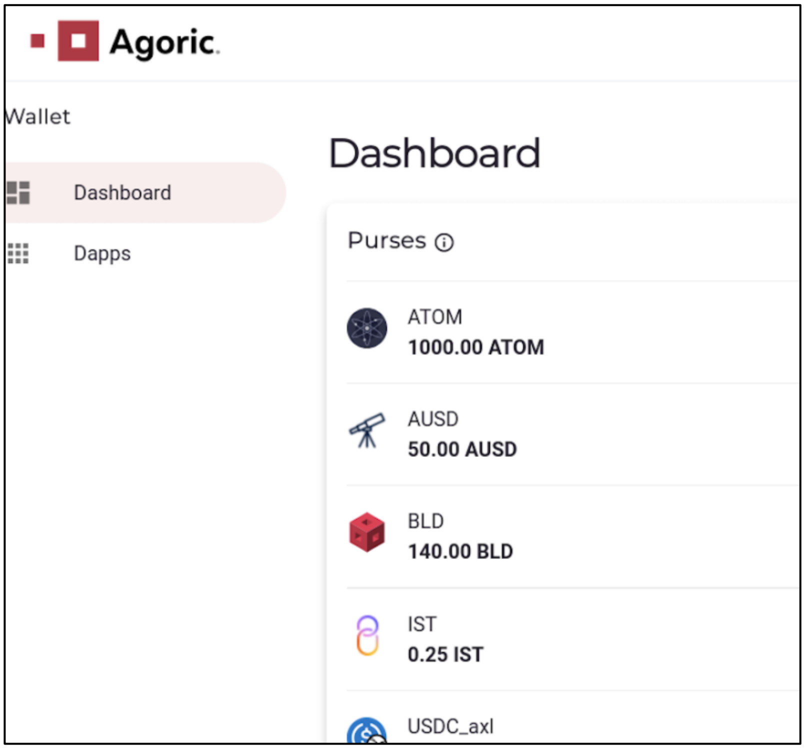
Severity: Low

Finding Overview

Users can submit offers via the Agoric CLI to open a vault. Each offer includes an `offerId` which is supposed to increment with each subsequent `offerId` submitted to the chain. However, if a user submits an offer with one `offerId` and then subsequently issues the same offer, collateral will be withdrawn, and a new vault will be created. This new vault will not be accessible via either the CLI or the Inter Protocol UI DApp which lists user vaults. Therefore, users will lose control over their collateral which is offered when creating the second vault.

Finding Detail

Using the same `offerId` across multiple vaults creates a new vault and leaves the vault inaccessible to the user. To begin with, we started with the `gov2` account which has 1000 ATOM.



ATOM
1000.00 ATOM

AUSD
50.00 AUSD

BLD
140.00 BLD

IST
0.25 IST

USDC_axl

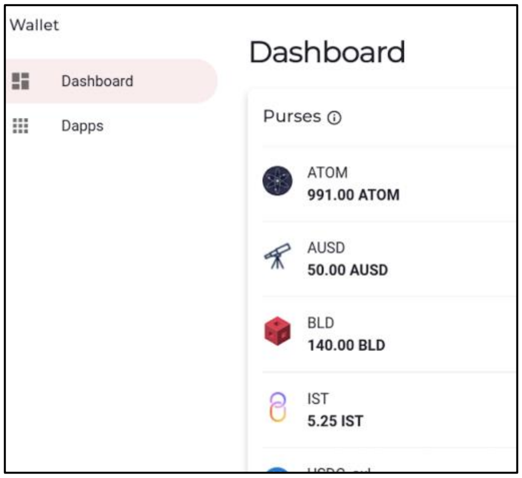
Gov2 User with 1000 ATOM

We create a new transaction below using the `agoric` command line utility. The `gov2` user offers up 9 `ATOM` as collateral, opens a vault, and can see only 991 ATOM is left in the smart wallet. The user can see this new vault in both the command line as well as the Inter Protocol UI DApp.

```
josh@vm:~/projects/agoric/agoric-sdk-841a7012831166f6c29f00bdffc907f2b35a7b0c$
OFFER=$(mktemp -t agops.XXX)
josh@vm:~/projects/agoric/agoric-sdk-841a7012831166f6c29f00bdffc907f2b35a7b0c$ agops vaults
open --wantMinted 5.00 --giveCollateral 9.0 >|"$OFFER"
running with options {
  offerId: 1681151307908,
  collateralBrand: 'IbcATOM',
  wantMinted: 5,
  giveCollateral: 9
}
josh@vm:~/projects/agoric/agoric-sdk-841a7012831166f6c29f00bdffc907f2b35a7b0c$ sendOffer
"$OFFER" gov2
Executing  [
  '--node=https://xnet.rpc.agoric.net:443',
  '--chain-id=agoricxnet-11',
  '--keyring-backend=test',
  '--from=agoric140dmkrz2e42ergjj7gyvejhzmjzurvqeq82ang',
  'tx',
  'swingset',
  'wallet-action',
  '--allow-spend',

'{"body":"{\\"method\\":\\"executeOffer\\",\\"offer\\":{\\"id\\":1681151307908,\\"invitatio
nSpec\\":{\\"source\\":\\"agoricContract\\",\\"instancePath\\":[\\"VaultFactory\\"],\\"call
Pipe\\":[[\\"getCollateralManager\\",[{\\"@qclass\\":\\"slot\\",\\"index\\":0}]],[\\"makeVa
ultInvitation\\"]]},\\"proposal\\":{\\"give\\":{\\"Collateral\\":{\\"brand\\":{\\"@qclass\\
":\\"slot\\",\\"index\\":0},\\"value\\":{\\"@qclass\\":\\"bigint\\",\\"digits\\":\\"9000000
\\"}}},\\"want\\":{\\"Minted\\":{\\"brand\\":{\\"@qclass\\":\\"slot\\",\\"index\\":1},\\"va
lue\\":{\\"@qclass\\":\\"bigint\\",\\"digits\\":\\"5000000\\"}}}}}}","slots":["board01547",
"board0257"]}\n',
  '--yes'
]
josh@vm:~/projects/agoric/agoric-sdk-841a7012831166f6c29f00bdffc907f2b35a7b0c$ agops vaults
list --from gov2 --keyring-backend="test"
published.vaultFactory.manager0.vaults.vault13
```

Creating a Vault Successfully



Dashboard

Purses

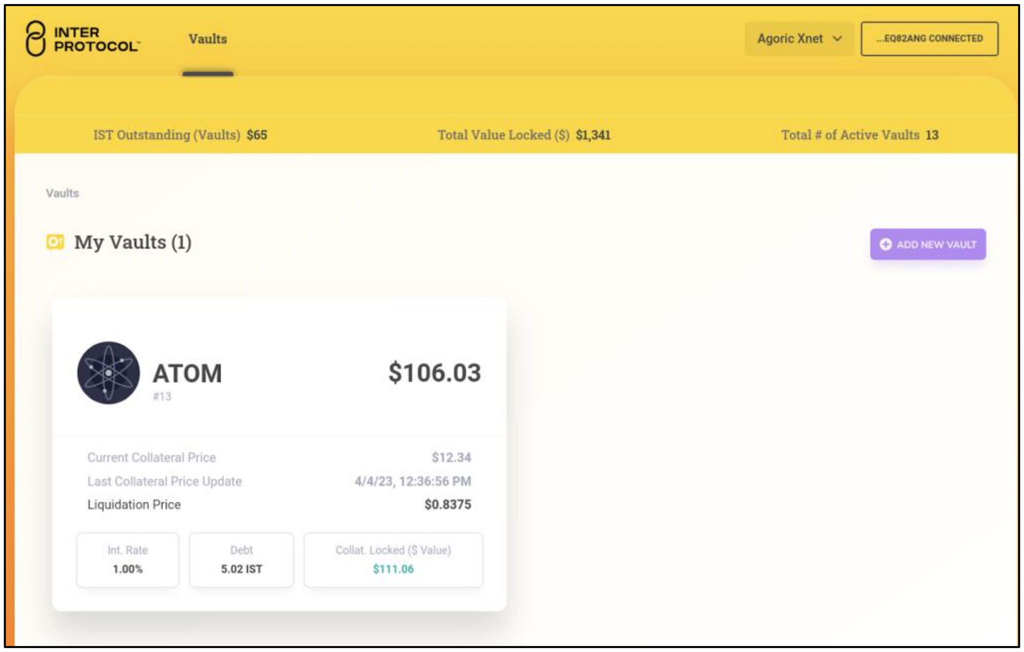
ATOM
991.00 ATOM

AUSD
50.00 AUSD

BLD
140.00 BLD

IST
5.25 IST

Gov2 ATOM Decremented by 9



IST Outstanding (Vaults) $65 | Total Value Locked ($) $1,341 | Total # of Active Vaults 13

Vaults

My Vaults (1)

ATOM $106.03
#13

Current Collateral Price $12.34
Last Collateral Price Update 4/4/23, 12:36:56 PM
Liquidation Price $0.8375

Int. Rate 1.00%
Debt 5.02 IST
Collat. Locked ($ Value) $111.06

New Vault with 9 ATOM at ~ $11 USD

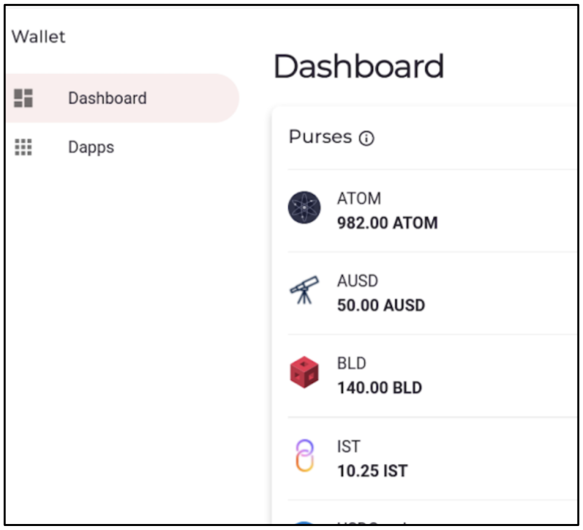
If a user creates a new vault with the same `offerId` as the previous vault, the new vault will not be visible to the user, even though the funds are collateralized and the vault is technically open.

```
josh@vm:~/projects/agoric/agoric-sdk-841a7012831166f6c29f00bdffc907f2b35a7b0c$ sendOffer
"$OFFER" gov2
Executing  [
  '--node=https://xnet.rpc.agoric.net:443',
  '--chain-id=agoricxnet-11',
  '--keyring-backend=test',
  '--from=agoric140dmkrz2e42ergjj7gyvejhzmjzurvqeq82ang',
  'tx',
  'swingset',
  'wallet-action',
  '--allow-spend',

'{"body":"{\\"method\\":\\"executeOffer\\",\\"offer\\":{\\"id\\":1681151307908,\\"invitatio
nSpec\\":{\\"source\\":\\"agoricContract\\",\\"instancePath\\":[\\"VaultFactory\\"],\\"call
Pipe\\":[[\\"getCollateralManager\\",[{\\"@qclass\\":\\"slot\\",\\"index\\":0}]],[\\"makeVa
ultInvitation\\"]]},\\"proposal\\":{\\"give\\":{\\"Collateral\\":{\\"brand\\":{\\"@qclass\\
":\\"slot\\",\\"index\\":0},\\"value\\":{\\"@qclass\\":\\"bigint\\",\\"digits\\":\\"9000000
\\"}}},\\"want\\":{\\"Minted\\":{\\"brand\\":{\\"@qclass\\":\\"slot\\",\\"index\\":1},\\"va
lue\\":{\\"@qclass\\":\\"bigint\\",\\"digits\\":\\"5000000\\"}}}}}}","slots":["board01547",
"board0257"]}\n',
  '--yes'
]
josh@vm:~/projects/agoric/agoric-sdk-841a7012831166f6c29f00bdffc907f2b35a7b0c$ agops vaults
list --from gov2 --keyring-backend="test"
published.vaultFactory.manager0.vaults.vault13
```

Re-sending the Same Offer

After sending the offer, the available `ATOM` is decremented by 9 again.



Dashboard

Purses

ATOM
982.00 ATOM

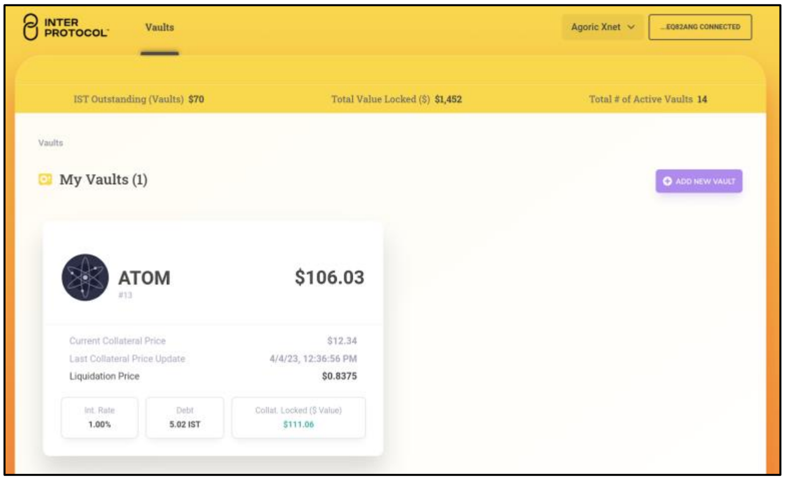
AUSD
50.00 AUSD

BLD
140.00 BLD

IST
10.25 IST

Gov2 Now Has 982 ATOM

However, the new vault does not show up in the browser application.



IST Outstanding (Vaults) $70 | Total Value Locked ($) $1,452 | Total # of Active Vaults 14

Vaults

My Vaults (1)

ATOM #13 $106.03

Current Collateral Price $12.34
Last Collateral Price Update 4/4/23, 12:36:56 PM
Liquidation Price $0.8375

Int. Rate 1.00%
Debt 5.02 IST
Collat. Locked ($ Value) $111.06

New Vault Not Visible

Recommendation(s)

Ensure that the `offerId` used when opening new vaults is unique. This could be done by appending a value generated by Agoric to the `offerId` provided by the user, or by ensuring the user knows to increment or otherwise alter the `offerId` used when creating new vaults.

References

CWE-20: Improper Input Validation

https://cwe.mitre.org/data/definitions/20.html

Limitations of Automated Liquidation

Severity: Info

Finding Overview

The Agoric chain relies on the oracle network to report price fluctuations for use in determining asset values. If the price of an asset drops to become worthless (a price of zero), and all oracles in the network report such a price to the aggregator, the aggregator will reject the price update. While unlikely, this is an effective limitation in the oracle network that may cause unexpected conditions within the Agoric chain. If the price will not update, vaults that currently house those assets as collateral will not be forced to enter liquidation.

Finding Detail

While testing, Atredis noticed that oracles that reported a price of zero for an asset were completely ignored by the vaults contracts and no attempt at liquidation would occur.

```
brandonperry@Brandons-MBP agoric-sdk % pushPrice 0.00 gov2 gov1

[snip]

brandonperry@Brandons-MBP agoric-sdk % pushPrice 0.00 gov1 gov2

[snip]
```

Pushing a Price of 0 Using Gov1 and Gov2 accounts

After waiting 45 minutes to an hour, Atredis did not see any vaults with reportedly worthless collateral attempt to be liquidated. Digging into why this occurred, the default contract terms require a value greater than zero.

```
const DEFAULT_CONTRACT_TERMS = {
  POLL_INTERVAL: 30n,
  maxSubmissionCount: 1000,
  minSubmissionCount: 1,
  restartDelay: 1, // the number of rounds an Oracle has to wait before they can initiate
another round
  timeout: 10, // in seconds according to chainTimerService
  minSubmissionValue: 1n,
  maxSubmissionValue: 2n ** 256n,
};
```

The Default Contract Terms Defined in `price-feed-core.js`

However, even if this configuration value were not defined explicitly, the number library used by Agoric may have also implicitly prevented a value of zero from being used as well. Since Agoric uses a library for enforcing that natural numbers are used throughout the smart contracts, a value of zero may be rejected since zero is not a natural number by definition. Atredis was unable to determine the veracity of this within the confines of this engagement however.

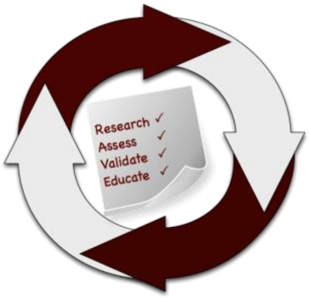
Recommendation(s)

As this is an accepted risk inherent to the platform, this type of behavior should be documented in the official public documentation as a limitation of the Agoric chain such that developers are aware of the limitations.

References

CWE-754: Improper Check for Unusual or Exceptional Conditions
https://cwe.mitre.org/data/definitions/754.html

Appendix I: Assessment Methodology



Atredis Partners draws on our extensive experience in penetration testing, reverse engineering, hardware/software exploitation, and embedded systems design to tailor each assessment to the specific targets, attacker profile, and threat scenarios relevant to our client's business drivers and agreed upon rules of engagement.

Where applicable, we also draw on and reference specific industry best practices, regulations, and principles of sound systems and software design to help our clients improve their products while simultaneously making them more stable and secure.

Our team takes guidance from industry-wide standards and practices such as the National Institute of Standards and Technology's (NIST) Special Publications, the Open Web Application Security Project (OWASP), and the Center for Internet Security (CIS).

Throughout the engagement, we communicate findings as they are identified and validated, and schedule ongoing engagement meetings and touchpoints, keeping our process open and transparent and working closely with our clients to focus testing efforts where they provide the most value.

In most engagements, our primary focus is on creating purpose-built test suites and toolchains to evaluate the target, but we do utilize off-the-shelf tools where applicable as well, both for general patch audit and best practice validation as well as to ensure a comprehensive and consistent baseline is obtained.

Research and Profiling Phase

Our research-driven approach to testing begins with a detailed examination of the target, where we model the behavior of the application, network, and software components in their default state. We map out hosts and network services, patch levels, and application versions. We frequently use a number of private and public data sources to collect Open Source Intelligence about the target, and collaborate with client personnel to further inform our testing objectives.

For network and web application assessments, we perform network and host discovery as well as map out all available application interfaces and inputs. For hardware assessments, we study the design and implementation, down to a circuit-debugging level. In reviewing source code or compiled application code, we map out application flow and call trees and develop a solid working understand of how the application behaves, thus helping focus our validation and testing efforts on areas where vulnerabilities might have the highest impact to the application's security or integrity.

Analysis and Instrumentation Phase

Once we have developed a thorough understanding of the target, we use a number of specialized and custom-developed tools to perform vulnerability discovery as well as binary, protocol, and runtime analysis, frequently creating engagement-specific software tools which we share with our clients at the close of any engagement.

We identify and implement means to monitor and instrument the behavior of the target, utilizing debugging, decompilation and runtime analysis, as well as making use of memory and filesystem forensics analysis to create a comprehensive attack modeling testbed. Where they exist, we also use common off-the-shelf, open-source and any extant vendor-proprietary tools to aid in testing and evaluation.

Validation and Attack Phase

Using our understanding of the target, our team creates a series of highly-specific attack and fault injection test cases and scenarios. Our selection of test cases and testing viewpoints are based on our understanding of which approaches are most relevant to the target and will gain results in the most efficient manner, and built in collaboration with our client during the engagement.

Once our test cases are validated and specific attacks are confirmed, we create proof-of-concept artifacts and pursue confirmed attacks to identify extent of potential damage, risk to the environment, and reliability of each attack scenario. We also gather all the necessary data to confirm vulnerabilities identified and work to identify and document specific root causes and all relevant instances in software, hardware, or firmware where a given issue exists.

Education and Evidentiary Phase

At the conclusion of active testing, our team gathers all raw data, relevant custom toolchains, and applicable testing artifacts, parses and normalizes these results, and presents an initial findings brief to our clients, so that remediation can begin while a more formal document is created. Additionally, our team shares confirmed high-risk findings throughout the engagement so that our clients may begin to address any critical issues as soon as they are identified.

After the outbrief and initial findings review, we develop a detailed research deliverable report that provides not only our findings and recommendations but also an open and transparent narrative about our testing process, observations and specific challenges in developing attacks against our targets, from the real world perspective of a skilled, motivated attacker.

Automation and Off-The-Shelf Tools

Where applicable or useful, our team does utilize licensed and open-source software to aid us throughout the evaluation process. These tools and their output are considered secondary to manual human analysis, but nonetheless provide a valuable secondary source of data, after careful validation and reduction of false positives.

For runtime analysis and debugging, we rely extensively on Hopper, IDA Pro and Hex-Rays, as well as platform-specific runtime debuggers, and develop fuzzing, memory analysis, and other testing tools primarily in Ruby and Python.

In source auditing, we typically work in Visual Studio, Xcode and Eclipse IDE, as well as other markup tools. For automated source code analysis we will typically use the most appropriate toolchain for the target, unless client preference dictates another tool.

Network discovery and exploitation make use of Nessus, Metasploit, and other open-source scanning tools, again deferring to client preference where applicable. Web application runtime analysis relies extensively on the Burp Suite, Fuzzer and Scanner, as well as purpose-built automation tools built in Go, Ruby and Python.

Engagement Deliverables

Atredis Partners deliverables include a detailed overview of testing steps and testing dates, as well as our understanding of the specific risk profile developed from performing the objectives of the given engagement.

In the engagement summary we focus on “big picture” recommendations and a high-level overview of shared attributes of vulnerabilities identified and organizational-level recommendations that might address these findings.

In the findings section of the document, we provide detailed information about vulnerabilities identified, provide relevant steps and proof-of-concept code to replicate these findings, and our recommended approach to remediate the issues, developing these recommendations collaboratively with our clients before finalization of the document.

Our team typically makes use of both DREAD and NIST CVE for risk scoring and naming, but as part of our charter as a client-driven and collaborative consultancy, we can vary our scoring model to a given client’s preferred risk model, and in many cases will create our findings using the client’s internal findings templates, if requested.

Sample deliverables can be provided upon request, but due to the highly specific and confidential nature of Atredis Partners’ work, these deliverables will be heavily sanitized, and give only a very general sense of the document structure.

Appendix II: Engagement Team Biographies

Shawn Moyer, Founding Partner and CEO

Shawn Moyer scopes, plans, and coordinates security research and consulting projects for the Atredis Partners team, including reverse engineering, binary analysis, advanced penetration testing, and private vulnerability research. As CEO, Shawn works with the Atredis leadership team to build and grow the Atredis culture, making Atredis Partners a home for some of the best minds in information security, and ensuring Atredis continues to deliver research and consulting services that exceed our client's expectations.

Experience

Shawn brings over 25 years of experience in information security, with an extensive background in penetration testing, advanced security research including extensive work in mobile and Smart Grid security, as well as advanced threat modeling and embedded reverse engineering.

Shawn has served as a team lead and consultant in enterprise security for numerous large initiatives in the financial sector and the federal government, including IBM Internet Security Systems' X-Force, MasterCard, a large Federal agency, and Wells Fargo Securities, all focusing on emerging network and application attacks and defenses.

In 2010, Shawn created Accuvant Labs' Applied Research practice, delivering advanced research-driven consulting to numerous clients on mobile platforms, critical infrastructure, medical devices and countless other targets, growing the practice 1800% in its first year.

Prior to Accuvant, Shawn helped develop FishNet Security's penetration testing team as a principal security consultant, growing red team offerings and advanced penetration testing services, while being twice selected as a consulting MVP.

Key Accomplishments

Shawn has written on emerging threats and other topics for Information Security Magazine and ZDNet, and his research has been featured in the Washington Post, BusinessWeek, NPR and the New York Times. Shawn is a twelve-time speaker at the Black Hat Briefings and has been an invited speaker at other notable security conferences around the world.

Shawn is likely best known for delivering the first public research on social network security, pointing out much of the threat landscape still exists on social network platforms today. Shawn also co-authored an analysis of the state of the art in web browser exploit mitigation, creating the first in-depth comparison of browser security models along with Dr. Charlie Miller, Chris Valasek, Ryan Smith, Joshua Drake, and Paul Mehta.

Shawn studied Computer and Network Information Systems at Missouri University and the University of Louisiana at Lafayette, holds numerous information security certifications, and has been a frequent presenter at national and international security industry conferences.

Nathan Keltner, Founding Partner and CTO

Nathan Keltner leads, executes and coordinates advanced, custom-scoped projects for Atredis Partners. Nathan’s primary focus includes hardware reverse engineering and penetration testing, red teaming, protocol analysis and private vulnerability research.

Experience

Nathan began his security career performing penetration tests and various security assessments for a large retail corporation, later expanding his career in consulting and specialization within red team penetration testing, exploit development, and software and hardware reverse engineering. Prior to starting Atredis Partners, Nathan most recently was a Senior Research Consultant on Accuvant’s Applied Research team.

Nathan has also worked extensively as a penetration tester, helping design penetration testing methodologies and workflows as well as leading complex red team, social engineering, and attack simulation engagements, as well as numerous reverse engineering and binary analysis projects.

Nathan’s research and exploitation assessments have recently focused on server hardware and embedded appliances, such as identification of vulnerabilities in BMC, UEFI, or OS firmware in related components. Previous expertise includes study of custom RF and ZigBee smart grid infrastructures, 802.15.4 and serial retail networks, multi-function ATM hardware and software, PIN entry devices, IPTV, VoIP hardware and software stacks, and modern networking access controls and identity management systems.

Key Accomplishments

Nathan has spoken at Black Hat USA, REcon, DEF CON, and other similar conferences on topics such as researching and exploiting smart grid radio frequency systems, exploitation in ARM TrustZone, advanced analysis of purpose-built system-on-chip architectures, and exploitation under limited-access user security models on the Windows platform.

Nathan holds a Bachelor of Business Administration degree in Management Information Systems from the University of Oklahoma, has held many information security and audit certifications over the years, and has been a frequent presenter at national and international security industry conferences.

Brandon Perry, Principal Research Consultant

Brandon Perry's responsibilities include complex web application and web services assessments, software reverse engineering, and source code reviews as well as red team and attack simulation engagements.

Experience

Most recently, Brandon was a Senior Penetration Tester at NTT Security, and contributed to the Foxglove Security blog at http://www.foxglovesecurity.com along with teammates (and now fellow Atredians) Justin Kennedy and Stephen Breen.

Prior to NTT Security, Brandon worked as a developer at Rapid7, supporting the Metasploit Framework and Nexpose vulnerability scanner. Brandon has also worked as a security engineer for Bethesda, working on AAA game titles such as DOOM, Fallout 4, and Elder Scrolls Online.

Brandon's extensive experience as a software engineer across several complex application stacks gives him key, practitioner-level insight into application security from the big picture down to the specifics of implementation and remediation. This development background makes Brandon uniquely suited for roles where interfacing and collaborating with development teams is crucial to a successful engagement.

Key Accomplishments

Brandon is the author of the No Starch book "Gray Hat C#", and the co-author of "Wicked Cool Shell Scripts, 2nd edition", and has spoken at DerbyCon and Infosec Southwest on web application vulnerabilities and exploit writing. Brandon has also heavily contributed to the Metasploit Framework with code and exploits, as well as several other open source security tools.

Joshua Domangue, Senior Research Consultant

Joshua Domangue specializes in delivering highly technical security assessments of web and mobile applications, evaluating embedded device security, and reviewing content protection mechanisms.

Experience

Joshua has over five years of experience in security consulting for clients spanning many industries. His experience includes application security assessments, source code auditing, evaluating embedded device security, and reviewing complex DRM systems.

Before Atredis, Joshua worked as a security consultant with Immunity, Inc., where he performed security consulting and research. Before that, Joshua worked at Independent Security Evaluators, where he performed application security analysis, reviewed DRM systems and content protection mechanisms, audited embedded systems for vulnerabilities, and served as the lead developer and maintainer for the SOHOpelessly Broken Capture the Flag contest which has been held at conferences such as DEF CON, DerbyCon, and others.

Key Accomplishments

Joshua holds a Bachelor of Science degree in Computer Science from the University of Maryland, Baltimore County (UMBC). He also holds the industry-specific Offensive Security Certified Professional (OSCP) certification.

Sara Bettes, Client Operations Associate

Sara Bettes assists the creation and completion of projects at Atredis Partners, ranging from the full pre-sales process to project design and management, to final delivery and follow-up. Her goals are to ensure all projects are executed in a way that reaches the goals of the client and assists the consultants at every turn.

Experience

Prior to joining Atredis Partners, Sara led a team that planned international sporting competitions, Olympic and national team qualifying events, as well as supported the mission of multiple non-profits. Her experience includes Live Sports Commentating, Staffing Management, Safety Plan Creation, Event Development, Public Relations, and Marketing efforts.

Key Accomplishments

Sara earned a bachelor’s degree in Mass Communications with an emphasis in Broadcast and Public Relations from Oklahoma City University.

Appendix III: About Atredis Partners

Atredis Partners was created in 2013 by a team of security industry veterans who wanted to prioritize offering quality and client needs over the pressure to grow rapidly at the expense of delivery and execution. We wanted to build something better, for the long haul.

In six years, Atredis Partners has doubled in size annually, and has been named three times to the Saint Louis Business Journal's "Fifty Fastest Growing Companies" and "Ten Fastest Growing Tech Companies". Consecutively for the past three years, Atredis Partners has been listed on the Inc. 5,000 list of fastest growing private companies in the United States.

The Atredis team is made up of some of the greatest minds in Information Security research and penetration testing, and we've built our business on a reputation for delivering deeper, more advanced assessments than any other firm in our industry.

Atredis Partners team members have presented research over forty times at the BlackHat Briefings conference in Europe, Japan, and the United States, as well as many other notable security conferences, including RSA, ShmooCon, DerbyCon, BSides, and PacSec/CanSec. Most of our team hold one or more advanced degrees in Computer Science or engineering, as well as many other industry certifications and designations. Atredis team members have authored several books, including *The Android Hacker's Handbook*, *The iOS Hacker's Handbook*, *Wicked Cool Shell Scripts*, *Gray Hat C#*, and *Black Hat Go*.

While our client base is by definition confidential and we often operate under strict nondisclosure agreements, Atredis Partners has delivered notable public security research on improving the security at Google, Microsoft, The Linux Foundation, Motorola, Samsung and HTC products, and were the first security research firm to be named in Qualcomm's Product Security Hall of Fame. We've received four research grants from the Defense Advanced Research Project Agency (DARPA), participated in research for the CNCF (Cloud Native Computing Foundation) to advance the security of Kubernetes, worked with OSTIF (The Open Source Technology Improvement Fund) and The Linux Foundation on the Core Infrastructure Initiative to improve the security and safety of the Linux Kernel, and have identified entirely new classes of vulnerabilities in hardware, software, and the infrastructure of the World Wide Web.

In 2015, we expanded our services portfolio to include a wide range of advanced risk and security program management consulting, expanding our services reach to extend from the technical trenches into the boardroom. The Atredis Risk and Advisory team has extensive experience building mature security programs, performing risk and readiness assessments, and serving as trusted partners to our clients to ensure the right people are making informed decisions about risk and risk management.