4/21/2018

Prepared by SHS Hundal and KA Thachek

Fog Technology LTD

Smart Contracting Quantum Fog

Toward a Secure and Robust Internet Project Plan

Contents

[2 Introduction 2](#_Toc512112333)

[2.1 Background & Rationale 2](#_Toc512112334)

[2.2 Project Proposal 2](#_Toc512112335)

[3 Phase I: Initiation 7](#_Toc512112336)

[3.1 Create Project Charter 7](#_Toc512112337)

[3.1.1 Define Project Goals 7](#_Toc512112338)

[3.1.2 Identify Methodologies 7](#_Toc512112339)

[3.1.3 Personnel Responsibilities 7](#_Toc512112340)

[3.1.4 Software Infrastructure Requirements 7](#_Toc512112341)

[3.1.5 Resource Procurement 7](#_Toc512112342)

[3.2 Create Project White Paper 7](#_Toc512112343)

[3.3 Create Project Yellow Paper 7](#_Toc512112344)

[4 Phase II: Planning 8](#_Toc512112345)

[4.1 Product Activities 8](#_Toc512112346)

[4.2 Scope Definition 8](#_Toc512112347)

[4.3 Create Scope Document 8](#_Toc512112348)

[4.4 Gather Requirement Specifications 8](#_Toc512112349)

[4.5 Prepare Research Instruments 8](#_Toc512112350)

[4.6 Investigate and Define User Requirements 8](#_Toc512112351)

[4.7 Investigate and Define Customer Requirements 8](#_Toc512112352)

[4.8 Draft Requirements Document 8](#_Toc512112353)

[4.9 Prepare Development Plan 8](#_Toc512112354)

[4.10 Create Development Plan Document 8](#_Toc512112355)

[4.11 Design System Flow Diagram 8](#_Toc512112356)

[4.12 Design Infrastructure Diagram 8](#_Toc512112357)

[5 Phase III: Execution 9](#_Toc512112358)

[6 Phase IV: Deployment 10](#_Toc512112359)

[7 Phase V: Operations 11](#_Toc512112360)

[8 Phase VI: Cycle Closure 12](#_Toc512112361)

# Introduction

## Background & Rationale

Data is the most important asset to any corporate or public organization. Preventing network intrusion and maintaining data integrity is expensive and failure to do so is often catastrophic. While there are several implementations available that out-source IT security features to a private cloud, oftentimes these are off-site data centers~~,~~ or secondary applications, using well-known encryption protocols such as SHA512 et al. It is important to note that these technologies can be easily breached by nefarious actors, namely cyber criminals who have sufficient knowledge and technical capabilities. In the case of off-site data centers, compliance and situational awareness are delegated externally to the organization, leaving the technical details of security protocols down to a black-box or a standard open source approach to various appliances that safeguard on-site technical infrastructure. Organizations that do not out-source these services and maintain such systems in-house are especially vulnerable.

## Project Proposal

Similar to Virtual Private Networks (VPNs), our approach builds on this by using a quantum fog computing platform to represent the domain network, which runs completely in-house via trusted nodes but in a decentralized peer-to-peer protocol. This includes all servers, workstations, as well as any remote clients and their applications. However, many enterprise environments do not have the ability to detect possible phishing attempts and are susceptible to a multitude of other cyber-attacks, thus giving rise to the prevalence of ransomware which encrypts sensitive data and holds the information hostage.

The single greatest risk is offered by end-users being incapable of distinguishing spoofed emails from legitimate sources, before it is too late. It is most common for end-users to accidentally click an unsecure link or open a malicious attachment, sent through email. All it takes is one breach of this kind to infect an entire network domain, as it will compromise system administrative privileges. This is a common trojan attack vector which acts like a worm, to detect all sensitive file formats and thereby encrypting them with a notice to issue payment in exchange for the decryption keys. In some cases, there is no guarantee that the keys will ever be turned over, regardless if payment is made or not. As such, there is rising demand for prevention of these intrusions, but they are not being addressed by conventional cyber-security standards.

We propose a software product using quantum neural networks, machine learning techniques, swarm intelligence optimization, as well smart contracts and blockchain compression technology to automatically detect unauthenticated actors connecting to the fog, deploying appropriate countermeasures, and safeguarding network security thus preserving data integrity and network stability. In this case, the implementation will act alongside current infrastructure on a predesignated port number, interfaced with all networked communications. Combining these technologies will offer multiple lines of defense against all known cyber-attack vectors, including but not limited to malware, ransomware, cross-site scripting, man-in-the-middle exploits, and cryptographic hash function reversals (password bypass attempts). This also minimizes IT resource requirements.

Implementation of the protocol described in qBitcoin (et al.), as characterized in (x), which provides a robust and simple interface, using Python modules. Similarly, this {compression, cryptographic} function provides an application layer, via Python, which allows it to be propagated over a peer-to-peer network, a la {Enterprise, University, Government etc.} networks.

It is of primary interest that such a protocol, leveraging quantum teleportation, is (a) cloneless, preventing replication or copying of the security software layer and (b) irreversible through proof-of-work which guarantees authentication for all data communications. Therefore, at any single point of entry, an attacker requires both the keys to the network, along with “Proof-of-Authenticity”, which refers to the smart contract and blockchain mechanism. The domain network sits inside of a digital sandbox, allowing the security framework to dynamically filter and prune the system for threats, isolating suspicious activity from normal events.

Using a combination of Python software packages, we are able to leverage modular effects to dynamically couple this proposed implementation against existing {Enterprise, University, Government etc.} environments. This provides a turnkey feature, with a front-end user interface that can exist across mobile, desktop, and server infrastructure through headless command-line integration {mobile application, headless command-line interface},This system connects directly to an {FTP, SALT, SSH, HTTP/HTTPS etc.} protocol, and propagates our functions over a private network, spanning multiple domains if necessary, and hence creating a secure fog which allows for the entry of fully authenticated nodes into the network.

Further

Module A (p2p):

Peer-to-peer networking is a well-characterized format of computing which distributes the available resources across the network in a decentralized manner to all available nodes, utilizing their combined computing power.

-Traditional, established, and tested thoroughly for a wide range of applications, including but not limited to software updates, autonomous transactions, and various network communication protocols.

-Sole purpose is to provide a persistent 'network warehouse', where both virtual and physical hosts can communicate in a file-sharing environment.

-Peers connected into the warehouse can only enter through proof-of-authenticity while rejecting impostors by recognition of spoofing patterns.

Module B (qnet):

-qnet (Quantum Network) holds information about each peer (e.g. node)

-the underlying proof-of-authenticity is a master equation embedded within the entire qnet

-using neural networks and bayesian inference, new peers connecting to the network are validated by a signature-based mechanism

-e.g. if a landing page, phish attack, ~~or~~ signature of a file, or object, shared over the qnet, is 'out of character', the qnet will throw an exception, and stonewall/alert/etc., flagging the attacker node for quarantine

- quantum teleportation provides the medium for transactions, where a transaction is an inference(x) or master equation. the quantum neural network will, like a traditional ANN, give no regards to the "how", only requiring inputs, mapped to desired outputs. Inputs and outputs are defined as follows:

input := authenticated user connecting to the local fog

QNN x

output := authentication at any given end-point (e.g. any node) is identical to the input

Using quantum teleportation mitigates the need for ledger or blockchain, a la bitcoin, ethereum, etc. that is, cancelable database computing. Once the information is sent to the QNN, it is impossible for a transmitter to keep hold of that data (reference to japanese guy with qcash)... the phase space that the guy captures is perfect for end-to-end transacting.

The combination of Module A and Module B represent one local quantum fog, whose network topology can be mapped and represented on an Intranet. Furthermore, ~~since~~ this representation is written entirely in Python and has constituents in Q#, Linux, Unix- and Windows-based environments, all of which can run this package natively on desktop and/or mobile environments, completely serverless, implying that node validation to the fog is representation-free, ~~vis a vis~~ which means it can run on any internet medium, including but not limited to 5G, Wi-Fi/Mi-Fi, and fiber optic environments, while offering unprecedented security, while operating on any network, domain, or protocol.

Why is this "quantum"?  
  
Peer-to-peer simulation hypothesis posits that there exists no central server, but instead each computer runs the software or protocol in parallel to each other. One special property defined for any particular measurement from a single computer will result in the appearance of collapsing wave-like dynamics, which will return a single deterministic measurement. This can be applied to any level of the network, and allows for a unified result serving as an event identifier, since all nodes are constantly listening and updating their status with each other with concurrency.

Extensibility: The topologies generated by the quantum fog can be used for the purposes of data compression, as well as knowledge discovery, and finding bugs in programs with the possibility of patching them automatically. The modular nature of our approach allows for new adapters to be implemented, allowing for the app to support new protocol types, as well as

Minimum Viable Product

Use Cases

Proposal or Value Proposition: This suite of quantum protocols purports to do the following

IF Data Compression:

Reduce data storage requirements by potentially 80%

IF Intrusion Detection:

Mitigate financial~~, temporal,~~ and personal damages due to breach, identity theft, or loss of credentials.

~~Corporate~~ Domain end-point user validation mitigates financial and personal damages due to unknown actors

IF Knowledge Discovery:

Bridge pre-theoretical notions from Quantum Neural Networking to the world of Business Intelligence. Provide analysts with a suite of tools to mitigate risk factors pertaining to real estate procurement, energy site probabilities, and others.

IF Decentralized Data Warehousing:

Provide corporate environments a new approach over 4G+ networks to store data, hence mitigating the need for rack space and overclocked cloud time.

Policy Implications: A full data migration from bare-metal cloud environments to a p2p interface, while possible, can be expensive, and still does not account for centralization on the network. Going this route would require the onboarding of a full “DevOps” team, including, but not limited to: database specialists who can recognize and interface corporate credentials, information, and cryptography to the new system,

Cost Effectiveness and Growth Potential:

Migrating to Local Intranets:

Mitigating Security Threats:

Compression:

# Phase I: Initiation

## Create Project Charter

### Define Project Goals

### Identify Methodologies

### Personnel Responsibilities

### Software Infrastructure Requirements

### Resource Procurement

## Product Development and Operations

### Create Project White Paper

### Create Project Yellow Paper

### Minimum Viable Product

# Phase II: Planning

## Product Planning Activities

### Scope Definition

#### Create Scope Document

### Gather Requirement Specifications

#### Prepare Research Instruments

#### Investigate and Define User Requirements

#### Investigate and Define Customer Requirements

#### Draft Requirements Document

### Development Strategy

#### Create Development Plan Document

##### Design System Flow Diagram

##### Design Infrastructure Diagram

##### Design Entity-Relationship Diagram

##### Design UML Diagram

### Test Strategy

#### Design Test Diagram

#### Test Plan Document

### Deployment Strategy

#### Design Deployment Diagram

#### Deployment Plan Document

### Operations Strategy

#### Design Operations Diagram

#### Operations Plan Document

#### Maintenance and Revisions

#### Define Acceptance Criteria

## Project Planning Activities

### Scope Management, Costs, and Schedule Planning

### Develop Project Plan

#### Task List

#### Forecasting Estimates

#### Network Diagram

#### Work Breakdown Structure

#### Schedule

#### Milestone Plan

#### Organization Plan

#### Resource Assignments

#### Budget

### Communications Planning

#### Develop Communications Plan

### Quality Control Planning

#### Design Quality Control Diagram

#### Quality Control Plan

### Procurement Planning

#### Develop Procurement Plan

### Integration Planning

#### Develop Integration Plan

#### Coordinate Integration Plans

# Phase III: Execution

## Product Execution Activities

### Design Strategy

#### Develop Use Cases

#### Design Data Architecture

#### Design Hardware Architecture

### Perform Impact Analysis

### Develop Migration Strategy

### Design Product

#### Design UI

#### Design Objects

#### Design Workflow

#### Design Rules

#### Design Technical Infrastructure

#### Design middle-ware tier

#### Design Database

#### Design Connectivity With Other Systems

#### Design Permissions

### Design Tests

#### Manual Testing

#### Design UI Tests

#### Design Object Tests

#### Design Workflow Tests

#### Design Rules Tests

#### Design Middleware Tier Tests

#### Design Database Tests

#### Design Connectivity With Other Systems

#### Design Permissions Tests

#### Design Automated Tests

#### Design Automated Tests for Functionality

#### Design Automated Tests for Performance

#### Design Alpha Test

#### Design Beta Test

#### Design User Acceptance Tests

### Create Design Documentation

### Development Management

#### Build Prototype

#### Demo Prototype

### Develop Training Strategy

#### Design Training Document

### Develop Documentation

#### End-User Documentation

#### Design Developer Documentation

#### Develop Publicity

## Project Execution Activities

### Product Development

#### Develop UI

#### Develop Objects

#### Develop Workflow

#### Develop Rules

#### Develop Middleware Tier

#### Develop Database

#### Develop Connections with Other Systems

#### Develop Permissions

#### Develop Migration Scripts

### Create System Documentation

### Develop Tests

#### Develop Unit Tests

#### Develop System Tests

#### Develop Integration Tests

#### Develop Environment Tests

#### Develop Migration Tests

### Create Test Documentation

### Create Build

#### Execute Testing

#### Execute Manual Tests

#### Execute Automated Tests

#### Execute Alpha Testing

#### Execute Beta Testing

#### Execute User Acceptance Testing

#### Analyze Test Results

#### Compile Test Statistics

### Versions Finalized

# Phase IV: Deployment

## Product Deployment Activities

### Version Approved for Deployment

### Deployment Management

#### Schedule Release

#### Schedule Training

#### Schedule User Training

#### Schedule Support Training

### Notify Change Control

### Rollout

#### Backup Databases

#### Migrate Data

#### Install Software on Web Server (Master)

#### Install Software on Desktop (Slave)

#### Conduct Validation Testing

#### Confirm Successful Rollout

### Announce Launch

### Publish Press Release

### Write Article for Newsletter and Blog

## Project Deployment Activities

### Manage Scope, Cost, and Schedule

#### Monitor Product Changes

#### Monitor Project Changes

#### Renegotiate Scope, Cost, and Schedule Commitments

### Communication Management

#### Implement Communications Methods and Strategies

#### Select Tools

#### Specify Protocols

#### Distribute Information

#### Status Reporting

#### Document Project Procedures

#### Resolve Communication Problems

### Quality Management

#### Manage Quality Assurance

#### Establish Product Standards and Procedures

#### Define Quality Criteria

#### Define Quality Assessment Methods

#### Establish Project Standards and Procedures

#### Establish Requirements Standards and Procedures

#### Establish Change Management Standards and Procedures

#### Establish Documentation Standards and Procedures

#### Establish Configuration Management Standards and Procedures

#### Establish Source Code Control Standards and Procedures

#### Establish Version Control Standards and Procedures

#### Establish Peer Review Standards and Procedures

#### Establish Coding Standards and Procedures

#### Establish Code Standards and Procedures

#### Establish Commentary Standards and Procedures

#### Establish Build Standards and Procedures

#### Establish Release Standards and Procedures

### Deploy Technical Infrastructure

#### Build Requirements Repository

#### Create Requirements Templates

#### Implement Requirements Database

#### Build Change Management Repository

#### Create Change Management Templates

#### Implement Change Management Database

#### Build Documentation Repository

#### Create Documentation Templates

#### Implement Documentation Database

#### Implement Configuration Management Database

#### Implement Source Code Database

#### Implement Version Control Database

# Phase V: Operations

## Product Operations Activities

### Configure Project Environment

#### Configure Development Environment

#### Configure Test Environment

#### Configure Unit Testing Environment

#### Configure System Testing Environment

#### Configure Integration Testing Environment

#### Configure Automated Testing Environment

#### Configure Performance Testing Environment

#### Configure International Testing Environment

#### Configure Alpha Testing Environment

#### Configure Beta Testing Environment

#### Configure User Acceptance Testing Environment

#### Configure Staging Environment

#### Configure Production Environment

### Select Tools

#### Select Development Tools

#### Select Testing Tools

#### Select Production Tools

### Quality Control Management

#### Monitor Product Quality

#### Analyze Defects Reported

#### Analyze Change Requests Submitted

#### Analyze Tests Failed

#### Analyze Requirements Revised

#### Conduct SME Focus Groups

#### Monitor Project Quality

### Perform Audits

#### Audit Requirements Repository

#### Audit Change Management Repository

#### Audit Documentation Repository

#### Audit Configuration Management Repository

#### Audit Source Code Database

#### Audit Version Control Database

#### Audit Peer Review Practices

#### Audit Coding Practices

#### Audit Code Commentary Practices

#### Audit Build Practices

#### Audit Release Practices

#### Audit Project Environment

## Project Operations Activities

### Perform Maintenance

#### Maintain Requirements Repository

#### Maintain Change Management Repository

#### Maintain Documentation Repository

#### Maintain Configuration Management Database

#### Maintain Source Code Database

#### Maintain Version Control Database

#### Maintain Project Environment

#### Maintain tools

### Implement Process Improvements

### Manage Risk

#### Monitor Risk

#### Specify Responses

#### Procurement Management

#### Identify Procurement Requirements

#### Perform make-or-buy Analysis

#### Determine Localized Internal Specifications

#### Prepare Request for Proposals (RFPs)

#### Review Proposals

#### Evaluate Bids

#### Evaluate Vendors

#### Sign Contracts

### Procure Materials

### Form Project Team

#### Identify Project Team Requirements

#### Select Staff Members

#### Recruit Consultants

#### Train Project Team

#### Develop Project Training Materials

#### Conduct Business Orientation

#### Conduct Technical Orientation

#### Conduct Internal Class

#### Conduct External Class

#### Adapt Team

#### Conduct Team Building Activities

#### Support Remote, Virtual, and Telecommuting Work

#### Reallocate Roles and Responsibilities

### Manage Integration

#### Update Project Plan

#### Update Communications Plan

#### Update Risk Management Plan

#### Update Procurement Plan

# Phase VI: Cycle Closure

## Product Cycle Closure Activities

### Maintenance Management

#### Perform Impact Analysis

#### Monitor User Acceptance

#### Monitor Performance

#### Monitor Defects

#### Resolve Training and Support Issues

#### Resolve Technical Problems

#### Establish Maintenance Procedures

### Analyze Statistics

### Gather Requirements For Enhancements

### Acceptance Management

#### Demo Product to SHs, SMEs, and FMs

#### Obtain Confirmation of Acceptance Criteria

#### Acceptance Confirmed

## Project Cycle Closure Activities

### Manage Scope, Cost, Schedule Closure

#### Create Final Version of Project Plan

### Quality Management Closure

#### Evaluate Product Quality

#### Analyze Change Requests Submitted

#### Analyze Customer Complaints Received

#### Analyze Tech Support Calls Received

#### Analyze Processing Errors

#### Analyze Data Errors

#### Conduct User Focus Groups

#### Create Product Quality Assessment

#### Evaluate Project Quality

#### Analyze Estimates vs. Actuals

#### Analyze Plans vs. Realities

#### Analyze Compliance with Standards and Procedures

#### Analyze Process Effectiveness

#### Create Project Quality Assessment

### Manage Risk Closure

#### Identify Future Risks

#### Specify Risk Responses

### Manage Procurement Closure

#### Close Out Contracts

#### Manage Integration Closure

#### Consolidate and Index Documentation

#### Create Summary Statistics for Historical Reference