Mathematical Mesh Part I: Architecture Guide

Mathematical Mesh Architecture

<series>draft-hallambaker-mesh-architecture

<status>informational

<stream>independent

<ipr>trust200902

<author>Phillip Hallam-Baker

<surname>Hallam-Baker

<initials>P. M.

<firstname>Phillip

<organization>Comodo Group Inc.

<email>[philliph@comodo.com](mailto:philliph@comodo.com)

<also>http://mathmesh.com/Documents/draft-hallambaker-mesh-architecture.html

The Mathematical Mesh ‘The Mesh’ is an end-to-end secure infrastructure that facilitates the exchange of configuration and credential data between multiple user devices. The architecture of the Mesh and examples of typical applications are described.

# Introduction

The Mathematical Mesh is an infrastructure that makes the Internet easier to use by providing better security.

Public Key Infrastructure

## Security is free

The traditional approach to security usability attempts to reduce the burden security places on the end user to the point that they are willing to put up with the inconvenience. This approach has failed and will continue to fail for one singular reason: most users are not prepared to accept any inconvenience for the sake of security most of the time. This is even true of security practitioners, most of whom rarely if ever use end-to-end encrypted email.

The only Internet security solutions that have achieved any measure of widespread adoption are those that are either inescapable if users are to do their work (VPNs, S/MIME email in government agencies) or demand no effort of the user whatsoever (Firewalls, TLS, SSH, Signal, Wire, etc.).

If an application is to be used securely, it is not sufficient that it be possible for a learned expert to use it in a secure fashion. Security must be the default and all the operations necessary to configure and maintain the secure configuration must be performed automatically.

Phillip Crosby, the quality engineering specialist coined the slogan 'quality is free'. If we are to secure the Internet, we must develop security technologies that impose no cost on users. Fortunately, there is a straightforward solution:

*Any set of instructions that can be written down and given to a user can be turned into code and given to a computer.*

The process of configuring a popular email application to use S/MIME security involves over twenty steps and took the author over quarter of an hour to complete. The tools provided by the Mathematical Mesh allow those same steps to be performed by a shell script requiring no user interaction whatsoever.

Automating the process of managing cryptographic configuration of applications is more than user convenience. Computers are designed to follow instructions precisely and reliably, users are not. User guides describing how to configure commonly used applications are frequently wrong, out of date or downright dangerous. A script signed by a trusted provider known to perform effective peer review is much more likely to configure an application correctly than a user attempting to follow instructions or a script the user writes for their personal use.

Public Key Infrastructure is a powerful and complex tool but so is an internal combustion engine. Modern cars are designed on the assumption that the driver will have precisely zero understanding of how they function beyond the need to add fuel, water and air into the correct holes from time to time.

Properly implemented, security is invisible to the user unless they have a specific security concern. An email application should provide end-to-end confidentiality and authentication automatically without the need for the user to lift a mouse click.

Badly implemented, security becomes a source of user frustration. The biggest source of frustration of all being the use of passwords to authenticate users.

## The user is not the device

## The user is not the product

## The devil is in the deployment

# Architecture

# Security Considerations

This document describes the security requirements that motivate the design of the Mesh.

# IANA Considerations

# Conclusions