AuthCodeSEC

extending EPP with
public key authentication to
make domain transfer faster and safer

Victor Zhou, Namefi.io

Motivation

- Enable specifying the gaining party
  - Gaining registrar to begin with, can potentially be registrant later
- Enable verification with cryptography
- Backward compatible with current practice

Feedback from IETF-121
And summary of our proposed revisions

Challenge about Domain Slamming (Scott and Rick)

- The importance of maintaining some friction in the domain transfer process to prevent "domain slamming"
- The **losing registrar must retain the ability to deny a transfer**
- Current ICANN policies are already moving towards making AuthCodes more secure (ephemeral, short-lived)

Our Revision to v01

- When AuthCodeSEC is cryptographically validated, losing registrar **SHOULD** be still able to decline domain transfer

Challenge about Crypto Signing (Jody)

- Jody noted most registrants cannot handle cryptographic processes
- Model A (registrant signing) would only work for:
  - Companies with large IT departments
  - Domain professionals with technical expertise
- Suggested Model B (registrar signing) might be more practical

Our Revision to v01

- Remove other signing Models, only keep Model B (registrar signing) for this RFC's scope.
- Also emphasize Registrar signing is **OPTIONAL**. Registrars who don't upgrade can fallback to use their current AuthCode, maintaining backward compatibility

Challenge about Algorithm

3. Technical Considerations
- Andy Newton appreciated the general idea of using public key cryptography
- Pavel Kovalik raised key management questions

Our Revision to v01

- **Propose to begin with using widely adopted cryptography in DNS such as RSA with SHA-256 etc.**
  - **Best compatibility for signing / exchange of public keys**
  - **Allow future extension of specifying other algorithms**

Design Question: Exchange of Public Keys

Would DNSSEC be ideal or do you have concerns if we suggest using DNSSEC as the primary/default mechanism for distributing public keys

- For losing registrar to **obtain public keys of gaining registrar**
- assuming most registrars now are more capable with managing DNSSEC (since a lot of them had to provide such features to their users)