MIT Bitcoin Expo, March 9, 2019

Secure Signatures:

Harder Than You Think

Andrew Poelstra

Director of Research, Blockstream

Secure Signatures

$$P = xG$$

$$\begin{aligned} R &= kG \\ e &= H(P, R, m) \\ s &= k + ex \end{aligned}$$

(s, R) is the signature.

$$P = xG$$

$$\begin{aligned} R &= kG \\ e &= H(P, R, m) \\ sG &= kG + exG \end{aligned}$$

(s, R) is the signature.

Secure Signatures

What makes a signature “secure”?

- If nobody (i.e. no probablistic poly-time algorithm) can extract the secret key from signatures?
- If nobody can sign a given message without the secret key?
- If nobody can sign *any* message?
- What if they're allowed to request signatures on other messages?
- The same message?
- What if they can change the key? Choose it freely?

- Also, does k *really* have to be uniformly random?
- Yes. But we can get away with setting $k = H(x \| m)$. Why?
- How about x?

Sign-to-Contract

$$P = xG$$

$$\begin{aligned} R^0 &= kG \\ R &= R^0 + H(R^0 \| c)G \\ e &= H(P, R, m) \\ s &= (k + H(R^0 \| c)) \quad + ex \end{aligned}$$

Sign-to-Contract

$$P = xG$$

$$\begin{aligned}
R^0 &= kG \\
R &= R^0 + H(R^0 \| c)G \\
e &= H(P, R, m) \\
sG &= (k + H(R^0 \| c))G + exG
\end{aligned}$$

Sign-to-Contract Replay Attack

Suppose $k = H(x \| m)$.

$$\begin{array}{rl} s = & (k + H(R^0 \| c)) + ex \\ - \; s = & (k + H(R^0 \| c')) + e'x \\ \hline 0 = & H(R^0 \| c) - H(R^0 \| c') + (e - e')x \end{array}$$

So we'd better have $k = H(x \| m \| c)$!

- If the hardware device knows c before producing R^0 it can grind k so that $(k + H(R^0 \| c))$ has detectable bias.
- If it doesn't know c how can it prevent replay attacks?
- Send hardware device $H(c)$ and receive R^0 before giving it c.
- Then $k = H(x \| m \| H(c))$.

Multisignatures

$$P_i = x_i G$$

$$P = \sum P_i$$

$$R_i = k_i G$$

(exchange R_i's)

$$R = \sum R_i$$

$$e = H(P, R, m)$$

$$s_i = k_i + e x_i$$

(exchange s_i's)

$$s = \sum k_i + \sum e x_i$$

Multisignatures

$$P_i = x_i G$$
$$P = \sum P_i$$

$$R_i = k_i G$$

(exchange R_i's)

$$R = \sum R_i$$
$$e = H(P, R, m)$$
$$s_i G = k_i G + e x_i G$$

(exchange s_i's)

$$sG = \sum k_i G + \sum e x_i G$$

Multisignatures

What does it mean for a *multisignature* to be secure?

- Now the attacker can be a signer? Freely choose the key?
- How about *all* the signers? All but one?
- Start multiple signing sessions in parallel?

Multisignatures

- In fact the just-described scheme is insecure in multiple ways.
- Rogue-key attacks; if $P = \sum P_i$ then a bad signer can choose the whole key.
- So set $P = \sum \mu_i P_i$ where μ_i is “random”. (Hash P_i? Or *all* the P_i's?)
- Parallel attack: grind R's until you get a lot of e's that sum to each other.
- So add an extra round where everyone precommits to R_i, preventing any individual from grinding R.

Thank you.

Andrew Poelstra
clauspschnorr@wpsoftware.net