2022-02-08

Paper: An investigation of historic? cryptographic flaws in randomized censorship circumvention protocols

SS stream decryption oracle
- no auth
- same key both directions

obfs4 elligator2
- ~~history obfs2, obfs3~~

VMess something...
- but VMess often encapsulated

~~hybrid SS UDP thing?~~

- generally, indist from random
- "look like nothing" actually looks like something, just not like anything else
- why effective? it's a mystery, but they are
- so much so that markets, e.g. airports exist

[not about traffic analysis, WF, Frolov disconnection.

- Why aren't these flaws devastating in practice?
  - through a purely cryptographic lens is not the right way to view these protocols.

circ developers work in an environment where they have to constantly adapt anyway. agility more valuable than up-front design. Users are accustomed to adaptation and disruption, indeed that's the threat model.

student man quote

---

address blocking is a bigger concern, even w/ perfect crypto.

Modeling censors difficult, delayed responses
- cite Tsai

Tschantz poly/stego. Depends how you model censors: "permit what is not denied" "deny what is not permitted" evidence says that the former predominates. Censors are concerned with the collateral costs of overblocking, ("constrained")

crates.io/ristretto
montgomery.rs elligator_encode
draft-irtf-cfrg-hash-to-curve-10 §6.7.1 §68.2

Roberts ~~and~~ on friction as "tax" on access.
SS not a great protocol but fast, reliable, stable.
Censored users optimize for their own requirements.

Upgrade progress from meskto

Why care about uniform randomness at all?
Why is that a goal?
It's a fair question.
Why is a censor more likely to block 255 random bits
than 256 random bits?
Short answer: it works empirically
If it were easy to detect + block, presumably they
would have done so?
Can guess at ~~the~~ reasons: look like nothing in particular,
risk of false positives look like a random stream, insofar as that
can be characterized — and perhaps that's the rub.
If you want to say, the censor can just do this or that,
consider what may be hidden in that "just".

otoh, it could be that the censor is just not trying
to very hard, or deliberately passes
evidence for increased blocking at certain times,
as if the censor had a "budget" of blocking
(can't block everything all the time)
or making a rational decision to spend limited resources on other things

don't know of cases of these bugs being exploited,
but also didn't try very hard.
(could check dates of obfs4 bridges)

long upgrade cycle

"Detecting Probe-Resistant Proxies" out of scope

- send draft to UPGen team