Testimony by
Ronald L. Rivest
(MIT Institute Professor, Cambridge, MA)

Venue:
Hearing held by
Committee on House Administration
Chairperson: Rep. Zoe Lofgren (California)

Hearing Title:
"Exploring the Feasibility and Security of Technology to Conduct Remote Voting in the House"

Date:
Friday, July 17th, 2020

Time:
1:00pm EST

Testimony (both oral and written):
==========

Chairperson Lofgren and members of the Committee: I thank you for inviting me to testify regarding the feasibility of using technology for conducting remote voting in the House.

My bottom line is that such remote voting is feasible and can be made adequately secure.

By way of introduction, I am an Institute Professor at the Massachusetts Institute of Technology; my background includes computer science, information security, cryptography, and election security.

I am well known for the invention, with Adi Shamir and Len Adleman, of the RSA public-key cryptosystem, the first (and still widely used) implementation of public-key cryptography, enabling both secure communications and digital signatures.

I have worked for over two decades on voting security.

I was a member of the Technical Guidelines Development Committee from 2004--2009, advisory to the Election Assistance Commission; I chaired the subcommittee on Computer Security and Transparency.

I am a founding member of the CalTech/MIT Voting Technology Project.

And I am on the Board of Verified Voting, a non-profit promoting voting system security, especially through the use of risk-limiting audits.

I speak here only about the security aspects of remote voting, not about the appropriateness of remote voting for the House; that question is beyond my pay grade!

I see that the House, under Resolution 965, is already using proxy voting for remote voting. That resolution also authorized the examination of ways to vote remotely in a secure manner; hence today's hearing.

As noted, I think the House is in a good position: there are indeed suitable secure voting technologies available.

The important reason why that is true is that House votes are NOT SECRET. Voting in the House is not based on secret ballots.

That makes all the difference, as manipulation or alteration of votes can be detected and corrected.

For the record, I note that in the US, SECRET ballot voting was first implemented in Massachusetts in 1888. However, implementing secure secret ballot remote voting is still beyond the state of the art.

Designing a secure voting system requires, first of all, a clear statement of the security objectives. A system can't be said to be secure if there is no specification of what security should mean for that system. What are the baseline voting security requirements? Here are four:

(1) Only eligible voters can vote, at most once each.

(2) Votes are cast as intended.

(3) Votes are collected as cast.

(4) Votes are counted as collected.

Each property should not only be true, but be VERIFIABLY true.

Counting (tabulation) is not an issue, since non-secret ballots can be posted publicly and the tally then verified by anyone.

One recommended principle for achieving voting system security is that of SOFTWARE INDEPENDENCE, a notion developed by John Wack and myself. This principle basically says that you never want to be in a position where you have to say, "Well, the result must be right, because the computer says so!"

In other words, the election outcomes must be AUDITABLE.

Here a sketch of a simple architectural approach for secure remote non-secret voting, to illustrate:

-- there is a public web site where all cast votes are posted

-- each congressperson composes his/her vote, digitally signs it, and sends the resulting digitally signed ballot for posting on the public web site.

Many digital signature schemes are available; NIST has developed digital signature standards. Digital signatures are now implemented in every browser. One approach uses the RSA public-key cryptosystem.

A nice thing about digital signatures is that the signature on a digitally signed document (such as a ballot) is verifiable by anyone.

Note that a digital signature is not just a cut-and-paste image of a handwritten signature; it is a mathematical function of the message being signed and secret information specific to the signer.

Digitally signed ballots can be authenticated using public information, both as to origin (who the voter is) and as to content (what the ballot says).

Vote manipulations are not possible, as forging digital signatures is not feasible.

The most an adversary can do is to delete or duplicate votes.

An adversary can conceivably delete or duplicate votes even now, with proxy voting. If a congressperson can't submit a ballot, they can't vote. Detection and correction mechanisms can work for voting with digitally signed ballots as for proxy voting.

It is important to note that voters (in this case congresspeople) can check, or audit, that their votes are correctly recorded on the public web site. Missing votes can be restored.

The approach sketched here bears many similarities to your current "proxy voting" procedures; the public web site becomes the "proxy" for those voting remotely. Indeed, such a system should provide a smooth and secure extension of your current proxy voting procedures, which need not be abandoned.

This sketch is intended only to show that it is possible to use technology to do remote non-secret voting in a secure manner; other approaches are possible.

This concludes my testimony; I would be happy to answer any questions you may have.