Helios

Search this site

Attacks and Defenses

This page documents known attacks against Helios and briefly analyzes their severity against various versions of Helios.

Mario Heiderich, October 2011

Attack Description: XSS attack on election name and description that shows up only in the booth (not in the election view page)

Analysis: a significant issue that needs to be addressed as it endangers sessions of administrators and sessions.

Solution: this bug was patched in github and deployed to https://vote.heliosvoting.org on October 4th, 2011.

Estehghari-Desmedt, August 2010

http://www.cs.ucl.ac.uk/staff/y.desmedt/slides/Hacking-Helios2.pdf
http://www.usenix.org/event/evtwote10/tech/full_papers/Estehghari.pdf

Attack Description: A browser rootkit is installed via an Adobe Acrobat buffer-overflow. This rootkit is Helios-specific and corrupts an existing Firefox extension to alter the displayed information to the user. The user's vote is flipped, and all verification channels are corrupted in that browser so that the flip goes undetected. The attack is demonstrated against Windows XP with Adobe Reader 7.0.0/8.1.2/9.0.0. Though these have been patched, the general approach would likely work using any newly discovered security hole in Acrobat or other.

Analysis: This type of attack was specifically highlighted in the first Helios paper: Helios does not attempt to defend against a corrupt web browser. Helios should not be used in elections where it is reasonable to assume that an attacker would be capable and willing to write a virus that infects voting members' machines.

Mitigation: Helios v3 allows audited ballots to be posted to the bulletin board so they might be checked by a different, hopefully

non-corrupt device. This is a small mitigation, and users should not assume that Helios is secure against a corrupt web browser.

Does it matter to me? If a corrupt web browser is part of your reasonable threat model, YES. You should not use Helios or other online voting systems. If you're more concerned about insider attacks and server security, then NO, this attack probably doesn't matter to you.

Wikström and Smyth-Cortier, December 2010

http://www.di.ens.fr/CryptoSeminaire.html#Attacking_ballot_secrecy_in_Heli

Attack Description: an attacker can take an existing vote, copy it, and cast it as his own. The vote can be modified in various ways so as to make it harder to detect that it is the same vote: introduction of spurious spaces in the JSON data structure, etc. Most problematic, because the proofs of knowledge of the vote plaintext are malleable -- because the challenge is computed with not enough parameters --, it is possible for an attacker to copy and alter a vote in a way that is effectively impossible to detect. Once it is possible to copy votes, then there is the potential of compromising a voter's privacy by copying their vote a sufficient number of times that the pattern is then detectable in the result.

Analysis: this is an interesting attack that would be particularly problematic if individual ballots were revealed in the end (i.e. in a mixnet setting). In a homomorphic setting, the risk of this attack being successfully carried out is low, as it requires "wasting" a number of votes to compromise the privacy of one voter.

Resolution: this problem is fixable by (a) ensuring that none of the components of the ballot are malleable, (b) strictly canonicalizing the ballot structure before generating its hash, and (c) ensuring that no sub-components of ballots are copied. The fix will be introduced in Helios v3.1 as part of the ballot data structure redesign.

Does it matter to me? Most likely, this does not matter. It only matters in an election where a number of voters are willing to give up their vote in order to compromise the privacy of ONE voter.

Comments

You do not have permission to add comments.