REFERENCE SOURCE CODE IMPLEMENTATION

MEDIA FREQUENTLY ASKED QUESTIONS

APRIL 6, 2004

WHY ARE YOU RELEASING VHTi SOURCE CODE?

VHTi is based on the concept of transparent provable elections. Being good students of cryptography, we understand that security through obscurity is no security at all. It is important that voters have confidence that their votes are cast and counted as intended. This is achieved by an open, transparent electronic voting system.

Between 1999 and 2002, VoteHere filed patents on the technology that underlies the VHTi protocols. In September 2003, detailed papers describing the VHTi cryptographic protocols were released. Now, VoteHere is releasing reference source-code that implements the cryptographic protocols.

WHAT EXACTLY ARE YOU RELEASING?

- Reference source-code that implements the VHTi technology
- Instructions on how to build the source
- Samples of VHTi's function usage
- Document of known issues

WHY ARE YOU RELEASING THIS NOW?

This has been a long process. Between 1999 and 2002, VoteHere filed patents on the technology that underlies the VHTi protocols. In September 2003, detailed papers describing the VHTi cryptographic protocols were released. Now, VoteHere is releasing reference source-code that implements the cryptographic protocols.

As we move toward integrating VHTi into electronic voting machines, we believe a public review process will bring critical transparency and openness to election technology. In addition, there have been several groups who have been critical of election technology and we believe that this level of transparency will help alleviate their concerns.

HOW DO THE PROTOCOLS AND API RELATE?

The API (application-programming interface) is software used to integrate the VHTi technology into a voting application. The VHTi technology is comprised of patent-pending cryptographic protocols. The API is a software representation of those protocols.

CAN THIS CODE BE USED TO BUILD AN ELECTION SYSTEM?

The VHTi API is ***not*** a voting system; it makes electronic voting systems verifiable. The VHTi protocols and reference implementation API are not freeware, shareware, or open source, they are proprietary and protected under patent, trademark, and copyright law.

CAN'T HACKERS USE THIS CODE TO LEARN HOW TO "HACK" THE SYSTEM?

Being good students of cryptography, we understand that security through obscurity is no security at all. Simple knowledge of the VHTi protocols cannot be used to "hack" the system.

WHAT IF REVIEWERS FIND ERRORS OR BUGS IN THE CODE?

VHTi ensures that election integrity is not dependent on software, including ours. Election integrity depends on the publicly disclosed election data, which can be

checked with VoteHere software or anyone else's that faithfully implements the VHTi cryptographic protocols.

Please send any errors or bugs found in this release to vhtifeedback@votehere.com.

IF A BUG IS FOUND, WILL YOU FIX IT?

We plan to address any issues raised by those who examine the code. This is a reference implementation. The mathematics behind the technology was released last September and anyone may write software that implements those algorithms.

WHO DO YOU EXPECT TO REVIEW THIS CODE?

There have been many groups critical of election technology due to lack of transparency. We hope they will take this opportunity to see that VHTi is a viable verification method.

DO YOU THINK OTHER ELECTION COMPANIES SHOULD RELEASE THEIR CODE?

In the presence of VHTi, there is no reason for other election companies to publicly release their source code. VHTi proves that (1) the voting machine isn't cheating or making mistakes and (2) provides for a meaningful recount -- even when faced with hackers, corrupt insiders, malicious software, and bugs. In the absence of VHTi, releasing their source code is a reasonable step.

WILL YOU BE RELEASING ALL VOTEHERE CODE?

This release currently represents all the VHTi source code. We will release any other code on a case-by-case basis.

WILL YOU PUBLISH THE FEEDBACK YOU GET FROM THE CODE REVIEWERS?

Reviewers are free to publish their own feedback. We will reflect input in revisions to the code where warranted.