Prepared Statement
Of
A Center for Correct, Usable, Reliable, Auditable, Transparent Elections (ACCURATE)

Prepared and offered by
Deirdre K. Mulligan
Acting Clinical Professor of Law
Director, Samuelson Law, Technology & Public Policy Clinic
UC Berkeley School of Law (Boalt Hall)
and
Joseph Lorenzo Hall
PhD Student UC Berkeley, School of Information (SIMS)

Before the
Senate Elections, Reapportionment & Constitutional Amendments Committee
Open Source Software –
Does It Have A Place In California's Electoral System?

February 8, 2006

Chairwoman Bowen and members of the Committee, thank you, for the opportunity to participate in this timely and important hearing. I and my colleagues Peter Neumann (SRI), and student Joseph Lorenzo Hall, are pleased to speak with you on behalf of ACCURATE, a interdisciplinary, multi-institutional research center funded by the National Science Foundation to improve the state of electronic voting systems.

As elections have become increasingly reliant upon complex technology, the job of the Secretary of State and all those involved in regulating, procuring and managing election systems has become increasingly difficult. The move to electronic voting has placed limits and barriers on the ability of election officials and the public to oversee election technology and ultimately elections. As computers replace paper and pen, the functionality of voting systems has moved from plain view to closed quarters. The previously transparent and familiar process of voting on pen and paper has been enclosed by technology that creates barriers to public and official knowledge and evaluation of the voting process. This "enclosure of transparency" occurs on several levels all of which must be addressed if states are to perform their obligations to ensure the integrity of elections. From the process which develops federal standards, to the entities that test voting technology, to the certification and review at the federal and state level, there is a

curtain that limits our ability to evaluate whether electronic voting technology is being held to appropriate standards, rigorously tested, and adequately prepared for the voting public. Secretaries of State, elected officials, parties, candidates, and the general public must be able to assess, at some level, and validate the trustworthiness of voting systems. To do so, requires windows into the lifecycle of voting technology through which independent evaluation of voting technology can occur.

In the role of Chief Elections Officer, the Secretary of State has the solemn responsibility to administer and oversee the conduct of elections under the California Election code. A logical starting point for the conversation about the degree of "openness" required of voting technology is to ask what level of access, review, and openness of code is necessary to ensure that the Secretary of State can establish with certainty that election technology supports election values. This inquiry is the focus of our prepared statement.

California's Oversight of Voting Systems

California leads the nation in efforts to improve the reliability, accuracy, integrity and security of electronic voting systems. Reforms begun by former Secretary of State Shelley, continued by Secretary of State McPherson, and bolstered by the guidance and input from legislators, have placed California at the forefront of efforts to improve the capacity of state officials to exercise their obligation to ensure the soundness and integrity of California's voting systems. In conjunction with long-standing obligations and authorities established by the California Election Code and Constitution, these reforms begin to make real of the Secretary of State's mandate to ensure the fitness of voting systems and procedures.

The Secretary of State, in his role as Chief Election Official, is obliged to ensure the fitness of voting systems and procedures. The California Election Code establishes numerous obligations and responsibilities of the Secretary of State. The Election Code grants the Secretary of State the power to make regulations concerning voting systems,[1] as well as investigatory and subpoena power over alleged violations of those regulations.[2]

[1] CA Elections Code § 19100
[2] *Id.*, § 19102

Title 2, Division 3 of the Government Code lays out the responsibilities and duties of the Secretary of State to administer the provisions of the Elections Code, to see that elections are efficiently conducted and that state election laws are enforced.[3]

With respect to electronic voting systems, the Election Code directs the Secretary of State to: "establish the specifications for and the regulations governing voting machines, voting devices, vote tabulating devices, and any software used for each, including the programs and procedures for vote tabulating and testing. The specifications and regulations must address the suitability, secrecy and integrity of the election system. Additional obligations and sources of requirements are found in: the Voter Bill of Rights,[4] which establishes by implication the obligation to ensure that voters can cast secret ballots, free from intimidation,[5] in a language other than English,[6] vote with assistance[7] and be provided with the ability to obtain a new ballot if the voter has made a mistake;[8] and, in sections 2.5 and 7 of Article 2 of the California Constitution[9] which also establish the right of the voter to vote in secret and to have their votes counted. By implication these two sources oblige the Secretary of State to ensure that election technology accurately captures the voter's expressed intent, maintains the privacy of the voter throughout the process, and accurately counts the vote as cast.

To facilitate the Secretary of State's need to assess election systems, the Elections Code requires that an exact copy of the source code for all ballot tally software programs be placed in an approved escrow facility prior to its use, and that the Secretary of State develop criteria governing access to the escrowed code. Under this provision the Secretary of State should be able to provide his office, or contractors or advisors to his office, access to source code in order to test and evaluate ballot tally software.

Finally, to ensure that "defective, obsolete, or otherwise unacceptable" systems are removed the SoS is authorized to decertify voting systems and withdraw approval up to six months prior to an election (or on shorter timescales if the SoS can show good

[3] Cal Gov Code § 12172.5 (2004).
[4] Election Code § 2300
[5] *Id.*, §§ (a)(4)
[6] *Id.*, §§ (a)(8)
[7] *Id.*, §§ (a)(6)
[8] *Id.*, §§ (a)(5)(A)-(B)
[9] http://www.leginfo.ca.gov/const-toc.html

cause). When Kevin Shelley revoke approval of a variety of Direct Recording Electronic voting systems—on April 30, 2004[10] and was the subject of the *AAPD & Benavidez v. Shelley* (324 F. Supp. 2d 1120) lawsuit in the Central District of California in 2004 – he was acting under this authority.

Recent legislative measures and regulatory changes in California provide new tools and authorities to the Secretary of State to further his ability to fulfill his obligations to establish requirements and procedures to ensure the fitness of voting systems and procedures. The recently passed paper trail legislation, now codified as § 19250-19252 of the Election Code, ensures that the voting system captures voter intent independent of the rest of the operation of each voting system. New requirements promulgated by the Secretary of State's office[11] as well as the establishment, in his office, of the Office of Voting System Technology Assessment increase the robustness of the state certification process. The recent decision to require volume testing provides another testing method to identify faulty systems and improve the performance and integrity of voting systems—a method absent from the federal certification process. The Secretary of State is now authorized to have voting system source code evaluated by third parties of his choice. These independent evaluations will be an important part of identifying voting system vulnerabilities that slip through the closed, opaque federal certification and testing process.

Requirements for Effective Oversight of Voting Systems

There are several prerequisites to effective public oversight of voting systems. First, the Secretary of State requires full and unfettered access not only to the source code of electronic voting systems, but to all material relevant to an exhaustive evaluation, including system documentation, change logs, manuals, procedures, and training documents. Second, the Secretary of State must have the resources, expertise, and time

[10] Office of the Secretary of State, *Decertification of AccuVote-TSx Voting System* (April 30, 2004) available at: http://www.ss.ca.gov/elections/ks_dre_papers/decert.pdf and *Decertification and Conditional Certification for certain DREs* (April 30, 2004) available at: http://www.ss.ca.gov/elections/ks_dre_papers/decert1.pdf

[11] See "10 Voting System Certification Requirements", available at: http://www.ss.ca.gov/elections/voting_systems/vs_factsheet.pdf. Item 5 says, "In addition to depositing the source code in an approved escrow facility, each vendor must deposit a copy of the system source code and binary executables with the Secretary of State. The Secretary of State reserves the right to perform a full independent review of the source code.

necessary to understand, manipulate and test the materials and the machines seeking approval. Third, the Secretary of State must have an appropriate method of testing and evaluation that includes security ratings along multiple axes, threat analysis, code review, architectural review and penetration and parallel testing, usability (including accessibility) testing, and methods for identifying risks to voter privacy and equal participation. Fourth, the public must be provided, at least, with the exact certification tests performed and the system's performance.

As discussed above, California law and regulation is evolving to provide these prerequisites to public oversight in an incremental, thoughtful manner. While continued improvements are necessary, California is moving in the right direction.

The Potential Role of Open Source Software

Requiring voting technology to use open source software is one method for increasing code transparency—a subset of the requirements for public oversight outlined above. As the list above details, open source code does not address many other areas in which greater transparency is necessary to ensure the integrity of the election system. There are other options for increasing the Secretary of State's access to voting technology code in a manner that facilitates public oversight and accountability. The code escrow and independent testing currently required under California law, may provide opportunities for independent experts to examine and test election technology, however they stop short of providing broad public access to source code. North Carolina's recent decision to require voting system source code to be escrowed and specifically that the Board of Elections perform a variety of testing not conducted during the federal certification process is another effort to facilitate public oversight through code disclosure.

Open source software is software released under a license meeting the "open source definition"[12] which among other things requires access to source code, permits modification and distribution of source code with executable code. Open source software may or may not be the product of an open source development process. For example, the Australian Capitol Territory's Electoral Commission (ACTEC) contracted with a

[12] See: http://www.opensource.org/docs/definition.php

software firm, Software Improvements, to design and implement a voting system that would run on commodity equipment such as desktop computers and servers. This is the first and only experience with a fully open source voting system in a public election. While the system is open source, released under the GNU GPL, it was not developed under an open source development model. No developers external to Software Improvements were able to make submissions to the code base. In fact, an academic researcher found the one and only error, but although he provided a "patch" – a snippet of code that describes how to fix the problem – due to concerns about the integrity of the software, Software Improvements did their own analysis of the error and implemented and tested their own fix.

Because open source software is freely reviewable by any member of the public it can be subject to multiple, independent reviews, it can be tested with debugging scripts, put through all sorts of modifications and testing, irrespective of the will or ability of the code developers. In theory, and in the instance of software such as Linux, such open, robust review can lead to a strengthened code base, and to increased public and user confidence in the system's performance and attributes. However, the number and expertise of the individuals who choose to participate in the improvement of the code determine the outcome here. One can imagine a code base released under an open source license that received scant attention, scant attention from individuals with relevant expertise, or worse yet attention from those with expertise but the intent to subvert rather than improve the system. This example highlights an important potential dependency of the open source model – the participation of some set of individuals with appropriate expertise and a desire to improve the system. There is no guarantee that an appropriately skilled and engaged community will exist in all markets. Certainly the existence of efforts such as the Open Voting Consortium, ACCURATE, and the wealth of computer security experts who feel invested in improving electronic voting suggests that such a community exists in the voting space, however its important to note that without such a community it is unlikely that the full benefits of open source software can be achieved.

Open source software may also be attractive because of the numerous ways in which vendors' intellectual property concerns have hampered the desire of state officials

to review, examine and test voting systems. Intellectual property has been raised as an objection to source code escrow requirements, independent code reviews, independent testing, testing of add-on products to address accessibility obligations, among others. While it is unclear whether the intellectual property concerns all have merit, it is clear that intellectual property concerns, including copyright and trade secret concerns, are frustrating state efforts to ensure the integrity and fitness of election systems.

Given its natural alignment with the transparency prong of public oversight and its natural ability to void many intellectual property objections to oversight, the use of open source software deserves serious consideration. Serious consideration requires deep attention not only to the possible value of open source software, but to its viability given the current regulatory and market structures. The regulatory barriers to entering the voting technology market, while deficient in several respects, are economically quite high. The process of federal certification and testing alone—aside from software and hardware development, state testing etc.—runs several hundred thousand dollars a system. While providing room for open source software in the voting system market may make sense, it is quite possible that the production of a viable open source voting system will require a commitment of funds from a foundation, wealthy individuals, or perhaps a rethinking of the appropriate methods of investing in election system software by the state and federal government. It is possible that vendors may enter the voting systems market content on competing on service contracts—as Sun, IBM and Red Hat do with respect to Linux—however given the relatively low margins and limited voting system market this may not occur.

In addition, there are some features of open source software that voting systems maybe uniquely unable to benefit from. A distinguishing characteristic of open source software is the ability of users to modify it to address particular needs, concerns, or risks of a given environment. The flexible nature of the code serves the needs of various users to be able to customize and specialize software without undue cost or limitation beyond their own skill set or pocket book. In the context of voting technology, modifications are a bug, not a feature. Software versions are tested and certified and not to be modified without recertification. Thus the value of this flexible model appears quite limited in this

market.

It is clear that some ways that companies capitalize off of open source do not translate well to the voting systems market. For example, given the concerns and problems with networking in election systems, it would be difficult for a company, as Google does with search services, to make money off of running open source voting software remotely. Given the fact that any modified voting system software must be recertified at both the federal and state level, it would also be difficult for a company to optimize or customize open source voting software for their customers when they would have to have the resulting product recertified.

On the other hand, there are models that might be viable in the voting systems market. For example, a number of universities including UC Berkeley, Stanford and MIT recently decided that it didn't make sense to pay companies licensing fees for software that centralizes course website creation and management. They instead have started a project, SAKAI that they operate under a "community source" model. Any institution is free to download, use and modify the software, but if an institution would like to contribute features to the code base, they have to meet a certain set of requirements and pay yearly dues. This might be a model under which a consortium of states or jurisdictions collaborates to produce open source voting software and then pay yearly dues for administration and certification of the resulting voting system. Of course, there are questions specific to a model like this, such as who handles system integration and hardware that would need to be answered.

Source Code Transparency

Currently, source code of voting systems is not generally available for public scrutiny—in particular, for examination by impartial expert analysts. One method of increasing transparency of source code would be to require vendors to make source code and related information available for review by a panel of independent experts, not just by the Independent Testing Authorities or NIST. The independent experts making up a review panel should be given full and unfettered access not only to source code, but to all material relevant to an exhaustive evaluation, including system documentation, change logs, manuals, procedures, and training documents. The independent panel of experts

should be tasked with producing a public report stating and justifying their conclusions as to the security and performance of a voting system. The panel must present convincing evidence that the voting system as a whole meets its requirements for security.

Vendors should bear the burden and cost of providing evidence to an independent review panel that their voting product is safe, rather than inspection bodies bearing the burden to show the system is not safe. Election officials should not certify, purchase, or deploy voting equipment until independent security reviewers are confident that the technology will function as required.

The federal certification guidelines lack any provisions that would require vendors and the Independent Testing Authorities to open the certification process or source code to public scrutiny and understanding. Despite vendors' push-back due to potential revelations of trade secrets, protecting vendors' intellectual property must be accomplished in ways other than by sacrificing election transparency.[13] For example, experts can review certification results and source code under protection of non-disclosure agreements. Copyrights and patents owned or licensed by vendors to protect their intellectual property would still be fully enforceable. The use of open source can discourage theft of trade secrets between voting equipment vendors, as vendors will have to remove such secrets from their code base or agree to release any trade secret protection. It is accepted principle among computer security professionals that "security through obscurity" is neither secure nor obscure.[14] As an illustration, portions of Diebold's source code were leaked onto the Internet, despite attempts to keep it secret. Vendors should be put on notice now that they will be required to publish their source code by a specified year, in order to give vendors time to comply.

---

[13] *See* Daniel P. Tokaji, *The Paperless Chase: Electronic Voting and Democratic Values*, 73 FORDHAM L. REV. 1771, 1794 (2005) (Vendors have claimed that their software is a trade secret and thus have guarded against any attempts to make their source code publicly available (citing Michael Ian Shamos, *Paper v. Electronic Voting Records – An Assessment* § 3.2 (April, 2004), *at* http://euro.ecom.cmu/people/faculty/mshamos/paper.htm).)

[14] *See* Wikipedia: The Free Encyclopedia, *available at* http://en.wikipedia.org/wiki/Security_through_obscurity (last accessed Sept. 28 2005). *See also* Tokaji, *supra* note 13, at 1794 (Stringent limitations on access to source code severely diminishes the opportunity to expose vulnerabilities or malfeasance (citing Eric A. Fischer, Election Reform and Electronic Voting Systems (DREs): Analysis of Security Issues, Congressional Research Service Report for Congress, Order Code RL 32139 at 26 (Nov. 4, 2003)).)

Conclusion

A long-term goal of ACCURATE is to develop models that facilitate improved oversight of all aspects of voting and voting systems. Given the state of the voting systems market and regulatory process, there is much we can do to improve the transparency of voting systems and the processes we use to ensure that they count every vote as it was intended to be cast. While open source or disclosed source via incentives or mandate is one method of increasing transparency, it is important to stress that there are other alternatives and that we should move to increase transparency in well-understood increments. There is no doubt that many questions must be answered and further research conducted to fully understand the use of open and disclosed source code in regulatory contexts. Whether California chooses to pursue or facilitate open source software, we believe it is critical that ultimately all voting system source code, design documents and security analysis should be made available to the public. Greater government and public oversight over the testing and certification processes depends upon access to robust information. As an incremental step toward full public oversight, source code and related information must be available to review by independent experts. We applaud the efforts of the legislature and Secretary of State to move toward transparency and meaningful public oversight.

Thank you again for the opportunity to participate in this important hearing. We look forward to working with the Committee and the Secretary of State's Office on these issues.