

# Rapid Architecture-Based Election Technology Verification (RABET-V)

## Pilot 1

# Rapid Architecture-Based Election Technology Verification (RABET-V)

## Pilot 1

### Acknowledgements

CIS would like to recognize the following individuals and organizations for their support in developing the RABET-V process, executing this first pilot, and reviewing pilot outcomes. Their time and expertise were invaluable in completing this important work.

In addition, CIS would like to thank the Democracy Fund for its generous financial support in developing and piloting the RABET-V process and this document's development.

### RABET-V Development and Administration

John Dziurłaj [Architecture Lead](#)  
Mike Garcia [Research and Reporting Lead](#)  
Brian Glas [Process Lead](#)

Katy Owens Hubler [Program Manager](#)  
Jared Marcotte [Program Coordinator](#)  
Aaron Wilson [Program Lead](#)

**Pilot Technology Providers:** The Pilot Technology Providers donated internal time and provided access to their systems to execute the pilot.

VR Systems – Electronic Pollbook and Election Night Reporting  
KNOWink – Electronic Pollbook

**Steering Committee:** The RABET-V Pilot Program Steering Committee provided expertise and advice on all aspects of the program and pilot.

Aaron Wilson [Steering Committee Chair, Former Senior Director of Election Best Practices, CIS](#)  
Christina Adkins [Legal Director, Elections Division, Office of the Texas Secretary of State](#)  
Dr. Jay Bagga [Co-Director, Voting System Technical Oversight Program \(VSTOP\), Indiana State University](#)  
David Beirne [Director, Federal Voting Assistance Program \(FVAP\)](#)  
Dr. Bryan Byers [Co-Director, VSTOP, Indiana State University](#)  
Nikki Charlson [Deputy Administrator, Maryland State Board of Elections](#)  
Amy Cohen [Executive Director, National Association of State Election Directors \(NASD\)](#)  
Geoff Hale [Lead, Election Security Initiative, Cybersecurity and Infrastructure Security Agency \(CISA\), U.S. Department of Homeland Security](#)

Jordan Jarnagin [former Certification Specialist, VSTOP, Indiana State University](#)  
Manikantesh Kilaru [former IT Specialist, VSTOP, Indiana State University](#)  
Jerome Lovato [Former Director of Voting System Testing and Certification, EAC](#)  
Ryan Macias [Election Security Consultant, CISA](#)  
Mike Moser [Director, Bureau of Security and Technology, Pennsylvania Department of State](#)  
Jessica Myers [Director of Policy, Office of the Secretary, Pennsylvania Department of State](#)  
Don Palmer [Commissioner, Election Assistance Commission \(EAC\)](#)  
Richard Rydecki [Deputy Administrator, Wisconsin Elections Commission](#)  
Molly Timperman [Former Project Specialist, VSTOP, Indiana State University](#)  
Spencer Wood [Chief Information Officer, Office of the Ohio Secretary of State](#)

**Technology Advisory Committee:** The RABET-V Technology Advisory Committee provided subject matter expertise to assist in the refinement of the RABET-V process.

Dr. Michael Garcia [Technology Advisory Committee Chair, Senior Advisor for Cybersecurity, CIS](#)  
Joshua Bloch [Software Engineering Institute, Carnegie Mellon University](#)  
Mary Brady [National Institute of Standards and Technology \(NIST\)](#)  
Lisa Carnahan [NIST](#)  
Lauren A. Cooper [Cybersecurity Engineer, Software Engineering Institute, Carnegie Mellon University](#)  
David Garlan [Software Engineering Institute, Carnegie Mellon University](#)  
Brian Glas [Open Web Application Security Project \(OWASP\), Software Assurance Maturity Model \(SAMM\)](#)

Gordon Gillerman [NIST](#)  
Robert Gordon [Principal Technical Consultant, Akamai](#)  
Gema Howell [NIST](#)  
Daniel Plakosh [SEI Software Solutions Division](#)  
Mary M Shaw [Software Engineering Institute, Carnegie Mellon University](#)  
Jonathan Spring [Senior Member of the Technical Staff, CERT/CC, Software Engineering Institute, Carnegie Mellon University](#)  
Ryan Wagner [Software Engineering Institute, Carnegie Mellon University](#)  
Beau Woods [Atlantic Council](#)

# Contents

---

<b>1</b>	<b>Executive Summary</b>	<b>1</b>
<b>2</b>	<b>Pilot Program Recap</b>	<b>3</b>
2.1	Background	3
2.2	Challenges and Resolutions	4
2.2.1	Organizing Requirements and Efforts	4
2.2.2	Results for Multiple Audiences	4
2.2.3	Visibility of Results	5
2.2.4	Converting Best Practices to Requirements	6
2.2.5	Inventing an Architecture Review Methodology	7
2.2.6	Incorporating Usability and Accessibility	9
2.2.7	Documentation Challenges	11
2.2.8	Handling System Configurations and Variations	11
2.2.9	Developing Testing Tiers and Matrix	12
2.2.10	Conducting a Pilot During a Pandemic and Major Election Year	13
<b>3</b>	<b>Research Questions and Findings</b>	<b>14</b>
3.1	Time and Cost Implications	14
3.2	Pre-Review Assessment Implications	20
3.3	Technical Evaluation Implications	21
<b>4</b>	<b>Operational and Economic Models</b>	<b>23</b>
4.1	The Current Environment	23
4.2	A Better Way Forward	24
4.3	Roles in a National Verification Model	24
4.4	Potential Operating Models	26
4.4.1	The Trusted Verifier Model	27
4.4.2	The Verifier-Field Test Model	27
4.4.3	The Verifier+ Model	27
4.5	Potential Economic Models	28
4.5.1	Technology Provider High-Subscriber Low Model	28
4.5.2	Technology Provider Low-Subscriber High Model	29
4.5.3	Technology Provider Subsidization Model	29
4.5.4	Government Technology Provider Subsidization Model	29
4.5.5	Technology Provider Listing Fee	30
4.6	Overcoming Early Economic Barriers	30
4.6.1	Seeding Administrator Functions	30
4.6.2	Seeding Technology Providers	30
4.6.3	Seeding Approval Authority Changeovers	31
4.6.4	Offering the Subscriber Side of the Market for Free	31
4.7	Conclusion	31

---

<b>5 Recommendations</b>	<b>32</b>
<b>5.1 Preparation</b>	<b>32</b>
<b>5.2 Iterations</b>	<b>32</b>
<b>5.3 Process Steps</b>	<b>32</b>
<b>5.3.1 Human vs. Tool-Based Reviews</b>	<b>32</b>
<b>5.3.2 Incremental vs. Full Reviews</b>	<b>33</b>
<b>5.3.3 Combining Security Claims Validation with Verification Testing</b>	<b>33</b>
<b>5.4 Automation</b>	<b>33</b>
<b>5.5 Economic Model</b>	<b>34</b>
<b>5.5.1 Starting Up</b>	<b>34</b>
<b>5.5.2 Administrator Duties</b>	<b>34</b>
<b>5.6 A Final Word</b>	<b>35</b>

# 1 Executive Summary

Since 2019, the Center for Internet Security (CIS) has been working to fill gaps in the security of non-voting election technology. This began with the publication of our [Security Best Practices for Non-Voting Election Technology](#) guide and continued with the development and piloting of the Rapid Architecture-Based Election Technology Verification (RABET-V, pronounced “rabbit-vee”) process.

This is the final report for that pilot. If you are unfamiliar with the RABET-V process, we suggest you read our [white paper](#) that lays out the early framework for the program.

RABET-V is a unique approach to the verification of system security. Instead of employing a monolithic and lengthy approach to testing conducted after a system is fully developed or modified, it uses an iterative, risk-based approach that supports rapid product changes by design.

The risk estimate is based heavily on the product’s architecture and provider’s software development processes. Lower risk changes may be tested using streamlined testing methods while higher risk changes may require a more in-depth set of testing for verification across versions. This approach leverages and encourages modern software development, testing, and deployment practices. It also provides incentives for technology providers to design their products with stronger organizational processes and preferred architectures, as this eases the testing process by reducing the time and cost associated with verification.

To determine the viability and effectiveness of RABET-V, CIS conducted a pilot in 2020 with the assistance of a steering committee composed of election officials, a technical advisory committee composed of security experts, and two technology providers who supplied their products for testing. Prior to conducting the pilot, CIS worked with a group of stakeholders to develop a set of research questions, which were used to evaluate the RABET-V process and are provided, with responses, later in this report.

During the course of the pilot, the core team developed detailed procedures to perform each step in the RABET-V framework. These procedures were published as the RABET-V Program Description and were followed and updated throughout the pilot. Along with this report, we are publishing version 1.0 of the [RABET-V Program Description](#), which contains the updates made during the pilot.

Drawing from the pilot and interviews with the steering committee and other stakeholders, we present the following conclusions:

- 1 RABET-V is a viable process for non-voting equipment.** The pilot successfully evaluated two electronic pollbook solutions and one election night reporting solution.
- 2 We can evaluate architectures and use the evaluation to assess risk of changes.<sup>1</sup>** The pilot developed a rubric to measure architectural maturity from a security perspective and completed three reviews on three very different architectures.
- 3 We can evaluate software development processes and use the results to assess likelihood of positive security outcomes.** The pilot used an established process maturity model from OWASP and completed process evaluations with two different election technology providers.<sup>2</sup>

1 In this document the first person (I/we/our) refers to CIS, which acted as the creator and Administrator for the RABET-V pilot program and the coordinator for its associated committees.

2 The Open Web Application Security Project (OWASP) is a nonprofit foundation that works to improve the security of software. Among its projects is the Software Assurance Maturity Model, which is used within the RABET-V process to measure organizational process maturity.

- 4 **We can develop a testing matrix that prescribes different levels of testing based on the type of change, the architectural maturity score, and the process maturity score.** The pilot created a process to take the three risk-determining inputs and create a testing matrix tailored to each product. This matrix specifies one of three testing approaches for any given type of change to the product.
- 5 **We can re-evaluate new product versions more quickly for products with higher process and architectural maturity scores.** The testing matrix provides for cheaper and faster testing methods for products with high maturity scores.
- 6 **RABET-V can be managed by a central administrator with various activities conducted by external specialists.** CIS acted as administrator and contracted with specialists to perform activities such as process assessments, threat modeling, architecture models, and functional testing.
- 7 **RABET-V is compatible with multiple operational and economic models.** The pilot developed several operational and economic models that RABET-V could work with and validated these with the steering committee.

The initial RABET-V pilot successfully demonstrated that RABET-V is a viable process for non-voting equipment. From this pilot, we learned what a permanent operational program should look like, how it should be administered, and what it would take to run it. We believe two lines of effort will bring RABET-V into a full-scale production program serving U.S. elections:

- A second round of pilots that generate the necessary data, documentation, and tools to standardize the RABET-V process across any type of non-voting elections equipment.
- A developmental phase to finalize the operational and economic models that will ensure RABET-V can be adopted by state election officials and technology providers.

The need for RABET-V is now. The lack of consistent testing across states and the use of traditional lengthy and very expensive testing processes represents an opportunity to improve election infrastructure security. The current process has significant disincentives for both election officials and election vendors to upgrade election systems to incorporate the latest security upgrades and fixes at a pace that matches the evolving threat environment. RABET-V helps close this gap and balances multiple, and often competing, needs:

- RABET-V provides rigorous verification and testing that meet the needs of critical applications like those in the election environment.
- RABET-V provides a methodology and incentives for rapid development and deployment of product updates that have become standard practice in contemporary industries that are dependent on information technology systems.

The rest of this report is organized into sections:

- **Pilot Program Recap:** We provide a detailed recap of how the pilot was executed and discuss the variations from the initial plan to final execution.
- **Research Questions and Findings:** We present our answers to the research questions, which were identified prior to the pilot, and discuss our related findings.
- **Operational and Economic Models:** We define possible operational and economic models for RABET-V.
- **Recommendations:** We conclude the report with recommendations for follow-on RABET-V work and how to transition it into production.

All supporting material is found in appendices at the end of this report.

## 2 Pilot Program Recap

In this section, we provide a recap of the pilot program and discuss several of the challenges faced. For each challenge, we discuss the solution we implemented during the pilot or a proposal for future efforts.

### 2.1 Background

In 2019, the Center for Internet Security (CIS) began work to fill gaps in the security of non-voting election systems. There are both voting and non-voting types of election systems. A "voting system" is defined in the Help America Vote Act ([Public Law 107-252](#)). A "non-voting system" is any other election technology system used to administer an election. Some examples include voter registration databases, electronic pollbooks, and the websites of government election authorities.

First, CIS developed the [Security Best Practices for Non-Voting Election Technology](#) guide to provide a comprehensive set of security best practices. At the conclusion of that effort, CIS began working on a process to verify election technology against those best practices. This effort produced the concept of Rapid Architecture-Based Election Technology Verification (RABET-V).



After receiving input from various stakeholders in November of 2019, CIS developed the RABET-V framework and released it in a white paper titled [How to Improve Election Technology Verification](#) during the 2020 Winter Conference of the National Association of Secretaries of State.

Based on the RABET-V framework and with financial support from the Democracy Fund, CIS launched the first pilot of RABET-V in February 2020. The RABET-V Pilot Program was guided by a steering committee comprised of election officials, election technology providers, and other election infrastructure stakeholders. The steering committee included representatives from the states of Indiana, Maryland, Ohio, Pennsylvania, Texas, and Wisconsin. Representatives from the Election Assistance Commission (EAC), Cybersecurity and Infrastructure Security Agency (CISA), National Association of State Election Directors (NASED), and the Federal Voting Assistance Program (FVAP) further comprised the steering committee. Two technology providers volunteered their products and time to participate in the pilot. Knowink submitted their electronic pollbook, Poll Pad. VR Systems submitted their electronic pollbook, EViD, and their election night reporting solution. CIS also engaged with a group of technology experts who made up a RABET-V technology advisory committee.

The RABET-V Pilot Program first established a RABET-V Program Description. The Program Description is a detailed guide on how to run RABET-V. It was developed in early 2020 and was iteratively reviewed and modified by the core team and program steering committee as necessary.

Using the Program Description, the Pilot Program conducted initial iterations on the technology providers' products. Each initial iteration of RABET-V executed all activities resulting in the creation of maturity scores for each product and a testing matrix to guide future iterations.

## 2.2 Challenges and Resolutions

Throughout the pilot, the team encountered various challenges, both logistical and technical in nature. This section recaps those challenges and how the team addressed them during the pilot or proposes to address them in future RABET-V efforts.

### 2.2.1 Organizing Requirements and Efforts

During the pilot, it became apparent we needed a concise way to define and organize the RABET-V efforts beyond the process activities defined in the framework. We decided to define 10 security control families (sometimes referred to as security services) that are used throughout RABET-V to help evaluate the product. These families organize the requirements, architecture evaluation, verification testing, and results and are defined in the Program Description.

### 2.2.2 Results for Multiple Audiences

During the development of RABET-V, we learned that various states want different types of results from a central testing and verification program. Some states want a decision on whether the product version is acceptable or not. Other states want more details and raw results in order to make a decision themselves on whether the product version's security is acceptable or not. Based on our interactions, the larger, better funded states want to process the raw results and the smaller states prefer a decision be made by the verifier.

RABET-V attempts to address these various needs by organizing the results into three maturity indexes that reflect the results from the process assessment, architecture review, and the verification activities of the RABET-V process, and providing a single verification decision to serve as a baseline that can be used or ignored by a state election jurisdiction.

The three maturity indexes are the Software Development Maturity (SDM) Index, the Security Service Architectural Maturity (SSAM) Index, and the Security Service Capability Maturity (SSCM) Index. Each index is described below.

#### 2.2.2.1 Software Development Maturity (SDM) Index

The SDM Index score is measured by the RABET-V Process Review activity and indicates the maturity of the provider's software development processes for security and usability. The RABET-V SDM score is based on the [OWASP Software Assurance Maturity Model \(SAMM\)](#).

Maturity scores are provided for each of the 17 software development areas (15 SAMM areas plus Usability and Accessibility). The scores range from 0 to 3, where 3 is the best.



### 2.2.2.2 Security Service Architectural Maturity (SSAM) Index

The SSAM Index provides scores that indicate how well the product's architecture supports each security control family. This is a measure of the reliability of the security service and how isolated the security service is from other system changes. These maturity scores are measured during the RABET-V Architecture Review.

The SSAM Index provides a maturity score for each of the 10 security control families. The scores range from 0 to 3, where 3 is the best.

### 2.2.2.3 Security Service Capability Maturity (SSCM) Index

The SSCM Index identifies the product's current capability level measured across the 10 security control families and provides a maturity score for each family. The scores range from 0 to 3, where 3 is the best.

The scores are based on how well the product revision meets the security requirements set forth for each security control family. The requirements are pass/fail. Any assumptions made about the configuration or setup of the product are documented with the result.

## 2.2.3 Visibility of Results

Throughout the pilot, we discussed the question of the visibility of the testing results. There are many competing approaches to this and it is something that should be investigated in future RABET-V efforts. Our proposal defines four audiences: 1) the public, 2) subscribers, consisting of state and local election offices, 3) technology providers, and 4) partners, entities in the community with whom the Administrator may share some information about findings.

While we do not make a final recommendation on the visibility for each of these stakeholder groups, we suggest some options to consider in later discussions.

### Public

- RABET-V process documentation: this includes all of the information about how RABET-V is run
- Registered Technology Provider (RTP) name and contact information: we define the concept of an RTP in the Program Description
- The name and version number of each product and product revision that has completed or is currently going through the RABET-V process
- A high-level description of changes in each product version
- The verification decision for each product revision that has completed the RABET-V process
- For each product revision, the three maturity indexes

### Subscribers

- Test matrix and testing methods used for each product version
- Mid-level scores for maturity index; e.g., index scores across each of the 10 security services
- Verification results by security requirement

### Technology Providers

- Detailed testing notes and component level evaluations
- Architecture diagrams and other artifacts developed during the RABET-V process
- Reproduction steps for weaknesses or vulnerabilities found

- Detailed index scores from maturity indexes

## Partners

We envision a role for partners like the Cybersecurity and Infrastructure Security Agency (CISA) and other information-sharing partners. We believe the level of information may vary by partner, but generally includes things like summary and trend information about verification findings with the hope that this will inform national-level responses without revealing specific information about any given product or provider.

### 2.2.4 Converting Best Practices to Requirements

RABET-V is not specific to any set of requirements. In fact, it can be used with any appropriate security requirements and the Administrator may choose to apply different requirements in different environments, as described later in this section.

We choose to start with the best practices published as the [Security Best Practices for Non-Voting Election Technology](#). This guide was developed by CIS in collaboration with a community of election vendors, election IT staff, and other technology experts.

Best practices generally do not have the specific, measurable details that are necessary for conformance evaluations. In order to test against the best practices, we converted them into security requirements.

We also noticed that there was overlap between some of the security requirements and the process assessment. Since the requirements were used to measure the Security Service Capability Maturity, we removed the process-oriented requirements from the SSCM to avoid duplication.

Finally, we mapped each requirement to one of the 10 security service families and assigned each requirement a maturity level of 1, 2, or 3. The Security Service Capability Maturity Index is calculated from the number of applicable requirements that are verified for the product.

The applicability of the best practices will vary for each type of technology verified under RABET-V. For instance, some requirements specific to web services won't be appropriate for some election technology, while others, such as those specific to physical devices, may be appropriate for other election technology. Each type of election technology will have a set of requirements tailored to its function in the election environment. This applicability should be reviewed regularly and will be the portion of the RABET-V process that changes the most for different types of election technology.

The final set of requirements, along with their assigned security service family and maturity level, is found in the Requirements Master Workbook in the [Supporting Documents](#).

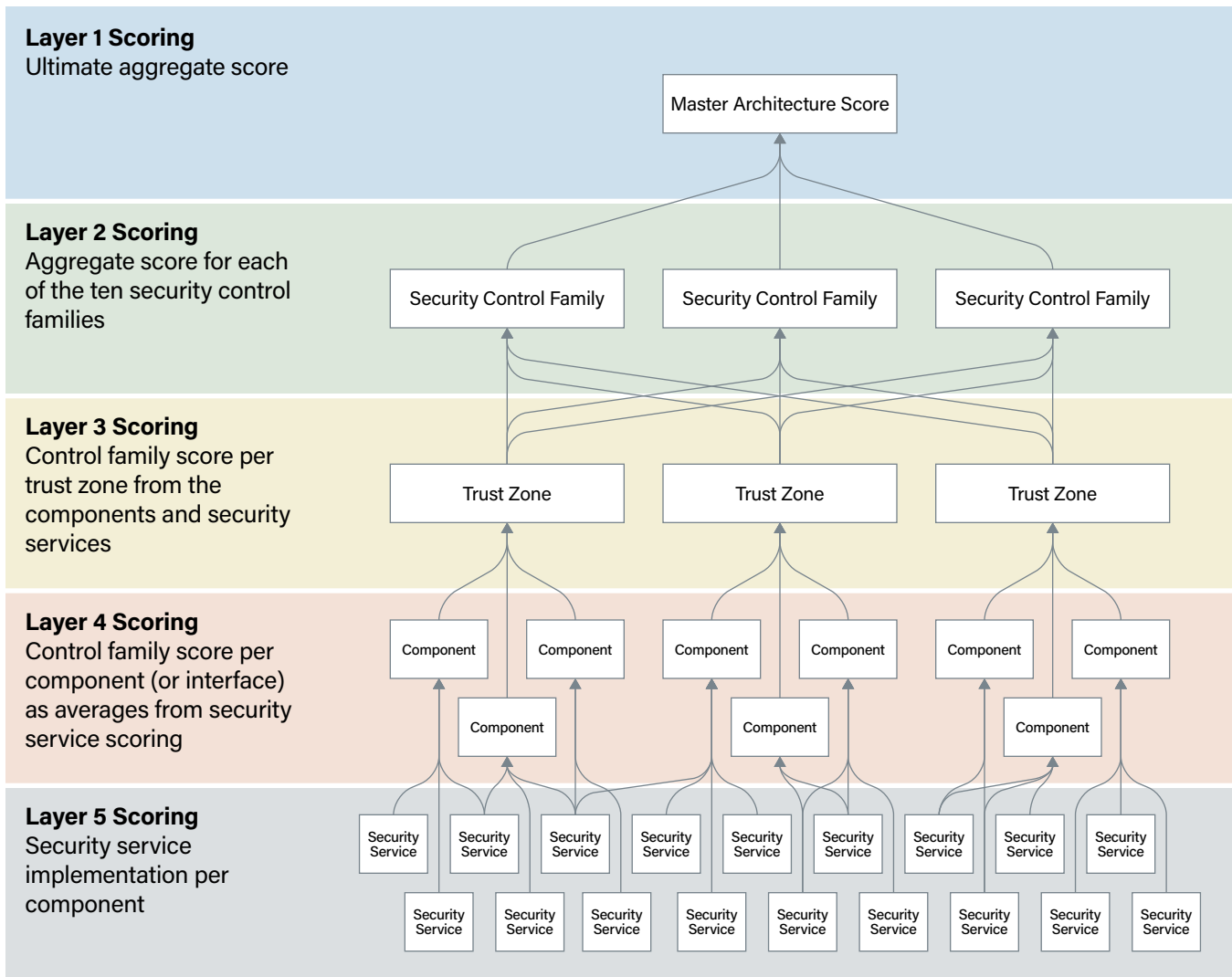
### 2.2.5 Inventing an Architecture Review Methodology

As the "architecture-based" part of the name indicates, RABET-V puts a significant emphasis on the product architecture with the intent for the architectural maturity to determine the level and the focus of testing performed on the product. Entering the pilot, we had broad buy-in from technical stakeholders that this was possible, but we did not have a method of determining architectural maturity for this purpose. While there are many architecture review processes, the team was unable to identify one that evaluated architectures to determine the risk of changes to security outcomes. This left the task to the project team.

Our first proposal, outlined in the initial Program Description, created two streams of effort. The first stream focused on system level analysis using threat modeling. The second stream relied on software analysis to identify and map software level components to security services. The initial approach used the system level analysis to inform the software level analysis and produced scores predominantly at the software level. We ended, however, with an approach where the software level analysis informed the system level and the scores were driven primarily from the system level. This is an important distinction that made the analysis much more reliable. This is most evident in the changes we made to the Security Service Architectural Maturity rubric. The initial version of the rubric discussed the *construction* and *usage* around security services and the language was written almost entirely from a software perspective. The final version is a much more system-driven approach, organized by reliability, maintainability, manageability, and depth.

In fact, one key decision made in the development of the final Security Service Architectural Maturity rubric was that better architectures rely as much as possible on dedicated and isolated components distinguishable at the system level. Realizing that this is not possible for all security services, the rubric has accommodations for the security services we believed required some level of configuration or integration with the solution's custom software. We called these *composite services*, an example of which is a Security Information and Event Management service, while those that we believe should be completely isolated we called *transparent services*, such as a firewall.

The rubric also considers the complex nature of systems with components and subcomponents. It does this by calculating scores at five layers, starting at the most detailed level of security service implementation per component or interface and resulting in a master architecture score. Then, the rubric has a unique category for *depth*, a top-down view over the whole architecture. In this manner, the architectural maturity score is a reflection of both bottom-up and top-down perspectives.



To support our analysis, we used three software tools. We used a software architecture management tool called Lattix®, a security and license compliance management tool called WhiteSource, and a software modeling tool called MagicDraw®.

Lattix performs analysis on the software solution using either source code or executable code and builds a dependency matrix. This illuminates how the system is constructed without having to perform direct source code analysis. We worked with each provider to identify the software modules related to each security control family, and then we analyzed the internal and external dependencies of each of those modules.

Separately, Lattix provides a change analysis capability. This highlights the software modules that have been updated from the prior version and can provide detailed listings of added, removed, and updated elements within each module. We did not use this feature in this initial pilot but believe it has great potential benefits in RABET-V for product iterations.

WhiteSource provided additional insight into the third-party dependencies of the systems. Specifically, it identified the third-party versions used and whether those versions were reasonably updated or had known security vulnerabilities.

MagicDraw was our selected modeling software to document the system architectures. MagicDraw component diagrams were useful in creating scores for various components. Initial setup was very time consuming but should be reusable as a template. We documented some of our usage of MagicDraw in several documents that can be found in the [Supporting Documents area](#) of this report: Understanding Architecture Review Diagrams and Scoring with MagicDraw. There is also a sample MagicDraw project called Sample Arch Review.

We believe there are many opportunities to improve the efficiency of the architecture review process but, even as executed, it proved a valuable and reliable method for understanding the quality of a product's design. In this pilot, we focused on developing a rubric and methodology that was sustainable and would produce consistent results. The next step is developing better instructions and tooling so that technology providers can prepare diagrams and perform some analysis themselves prior to initiating the RABET-V process.

### **2.2.6 Incorporating Usability and Accessibility<sup>3</sup>**

RABET-V is primarily intended to be a verification of product security. However, we heard early on from stakeholders that its value to the election community would be increased substantially if it also provided a measure of the usability and accessibility. We attempted to resolve this in the initial Program Description by expanding the process assessment to include usability and accessibility process maturity. The assessment establishes feedback loops as the basic level of maturity and more automated and more formal testing indicating higher levels of maturity. Each provider in the pilot was evaluated with this assessment.

While we think this approach is good for usability, we know it is lacking for accessibility. Due to the nature of accessibility requirements, we believe it is possible to incorporate actual evaluation of the accessibility of a product into RABET-V without changing the overall approach or timeline for conducting a verification. We also recognize that while the general RABET-V process can work for many types of non-voting election technology, evaluating usability and accessibility can vary greatly across types of technology. To this end, usability and accessibility requirements will have to be tailored based on the type of election technology and type of deployment—such as web-based deployments vs. device-specific systems.

We believe the approach to accessibility testing should use both automated and manual methods. Both are currently needed since even the best available automated tools identify only a small portion of known errors, produce false positives, and have difficulty assessing how well solutions are implemented. Functional testing with commonly used assistive technology must also be required. Similar to how we approach security, the manual methods will be used for the initial evaluation and for high-risk changes. The automated testing will be used for lower risk product updates.

<sup>3</sup> We would like to thank Dr. Diane Golden for her assistance with this section.

As an alternative to a risk-based approach, a sampling approach is also viable. As it will not be reasonable to audit everything, the sampling approach can be used to identify how much of an application must be reviewed for the accessibility audit. Although some automated tools can spider an entire site and give some useful information about accessibility, a certain sample of pages must still be selected for in-depth and manual testing. It is important that this sample is representative of content throughout the application, especially any interactive functionality with voters. If this approach is taken, the RABET-V Program Description must be updated to provide direction for how the access audit must be scoped so that a sufficient number of content and templates are reviewed to provide a representative sample of the overall platform and content (e.g., frequently used content like login, variety of content types, content with forms/tables/charts/graphics, interactive sites that use scripts or process user input). Most systems include both a template that holds the content and a database that populates the actual content to populate the template and create the page. When auditing these dynamic pages, it is important that the template and content, as well as the generated page, are audited. It's also possible to design a hybrid approach that includes risk-informed re-verification as well as sampling.

Another important factor to accessibility testing is the requirements for the accessibility auditor. As with other specialty areas of RABET-V, the accessibility auditing should be overseen by an organization or person with the requisite expertise and experience, addressing the attributes below.

### Experience

- The length of time that the vendor has been doing accessibility audits
- Whether accessibility audits are their primary business
- Whether they have experience with audits of very large online applications and their approach to sampling pages and prioritizing to ensure a valid review
- Whether they have conducted audits within the educational or instructional area

### Expertise

- The specific skills, education, and work experience of the auditor
- The specific software platforms and applications with which the testers have experience, such as programming expertise in accessibility, HTML, ARIA, and expertise with assistive technology and interoperability testing

### Testing Methods

- The specific automated and manual testing methods used to evaluate accessibility
- How testing is conducted with commonly used assistive technology
- Whether individuals with disabilities who use assistive technology are included in the audit as testers

### 2.2.7 Documentation Challenges

The initial Program Description outlined a basic set of documentation requirements. However, the core team avoided requiring the participating technology providers to develop new documentation to meet those requirements. This was done for three reasons: 1) we did not want to delay the start of the pilot to wait for new documentation, 2) we were looking for reasonable ways to reduce the burden on our participating providers, and 3) we were not confident we were asking for the right documentation. Instead, we took the documentation they developed for other certification efforts – such as those performed by the State of New York – and we mapped the documentation to our requirements.

In truth, this effort was not effective and being more prescriptive would not have helped either. Due to the nature of our efforts, we relied heavily on interview-based data collection techniques. We interviewed management, architects, developers, and others to collect the information we needed. Reviewing their documentation while conducting these interviews revealed a few things about documentation requirements:

- 1 Documentation will always be a lagging indicator of process or product maturity
- 2 The time spent reviewing documentation is equal to or greater than conducting interviews
- 3 Reviewers can often obtain greater depth about the process and product from interviews than from documentation

As a result, we have devalued traditional documentation requirements in the RABET-V Program Description with two exceptions:

- 1 User documentation. As opposed to technical documentation, user documentation is far more insightful for reviewers and its accuracy is a better indicator of product maturity. Reviewers find it helpful to provide context to product construction. We also believe it provides better return on investment for technology providers than technical documentation, the latter of which is difficult to keep updated, is often not used for technical evaluations, and has no value beyond the technical evaluation.
- 2 Visual, annotated architecture diagrams. During the pilot, we developed a series of architecture diagrams based on the data we collected primarily through interviews. These became the basis for our system understanding and the architectural maturity scoring. We have developed guidance on how technology providers can create these diagrams themselves. If the technology providers can present quality annotated architecture diagrams to RABET-V for verification and validation, this will greatly reduce the amount of time necessary to conduct an architecture review. Additional guidance and examples are necessary to fully equip providers to do this.

### 2.2.8 Handling System Configurations and Variations

During the pilot, it became apparent that each of the systems could be configured with various levels of security controls based upon the customer's requirements or preferences. This created a challenge for expedited evaluations. We certainly could not evaluate each variation during a pilot, nor would that be ideal in an operational version of RABET-V.

We determined the best approach was to have the technology provider define the specific configuration they wanted to present for verification. We decided to document the configuration choices and present those along with any scores. States and localities then could fully understand the context of the scores they were seeing and make any configuration modifications with those in mind. For example, if the technology provider chose to submit a less secure variation, the scores would be lower but provide more flexibility for the state or locality to make safe and secure configuration changes. If the technology provider chose to submit a more secure variation, the scores would be higher but would force the state or locality to increase their security risk with configuration changes.

This is an area that needs more exploration to determine the most effective way to handle multiple configurations of the same product.

### 2.2.9 Developing Testing Tiers and Matrix

RABET-V was developed on the concept that testing would vary based on the risk of the product changes made, thus allowing—and creating incentives for—smaller, more manageable changes. This meant that we needed to define testing levels and identify exactly how they would be applied. The chosen approach had to ensure that appropriate rigor was applied based on the risk created by the change.

To start, we began by developing testing levels. We decided to do this at the requirements level and vary the level of testing by varying the testing method used. We defined three testing tiers: *full*, *basic*, and *streamlined*. Testing methods included functional testing, data audit, penetration testing, configuration audit, documentation review, and artifact review. One or more testing method was assigned to each tier for each requirement. The definitions of the testing methods and their assignment to each requirement are found in the Requirements Master Workbook of the [Supporting Documents](#).

With the testing levels developed, we needed a consistent way to assign the testing level to the specific product revision. We decided that this needed to be done based on type of change(s) made to each security service, coupled with the relevant architectural and software development maturity scores. Based on the type of change(s) to each security service component(s), a testing tier would be determined for those security service requirements. For example, if there were low-risk changes made to the authentication components, the testing tier assigned would be streamlined. This means the streamlined testing methods would be used for each of the authentication requirements. The streamlined testing methods may vary for each requirement because certain requirements can only be tested in certain ways but, for each requirement, the process testing method chosen would be appropriate based on the low risk of the change.

To capture this, we built a list of change types and a master matrix that calculates risk from the change type, the process assessment score, and the architectural maturity score. This calculation results in a recommended testing tier. The master matrix and examples are available in the Security Service Testing Matrix Workbook of the [Supporting Documents](#).



There are several subtleties to this solution of testing tiers and a testing matrix. First, the approach encourages technology providers to develop and maintain robust automated testing capabilities. This is because many of the streamlined and basic testing tiers use the artifact review test method with many of the acceptable artifacts produced from automated tools or first-party testing; effectively the technology provider submits evidence of having conducted the tests themselves. If the technology provider cannot produce those artifacts, the more intensive testing tier is used. Second, the approach encourages providers to submit smaller product revisions. This is accomplished with the ordered change list we developed (with higher risk items appearing earlier in the list and requiring more extensive testing). RABET-V will select the change type most appropriate for each security service. If more than one change type applies, the higher one on the list is selected. This discourages bundling of changes into large releases and encourages smaller, more specific updates.

### **2.2.10 Conducting a Pilot During a Pandemic and Major Election Year**

As one reviews this pilot effort, it is important to recall that we conducted the pilot during a presidential election year and during a global pandemic. The challenges presented by the election were known ahead of time, and we understood that the operational demands on our participating technology providers and steering committee members would be priority over the pilot efforts. The pandemic further exacerbated these challenges by removing people from the typical work environments, forcing all interactions to be done remotely, and limiting the volume of people who could physically test the devices.

These challenges primarily impacted the pilot timeline, and not the effectiveness or reliability of actual results. We estimate that we lost three months to General Election activities between September and December, but we also saw impacts in the spring and summer as many states were delaying their primaries and shifting their voting models to more vote-by-mail approaches. These changes required significant efforts from our steering committee members and participating providers. In fact, we had interest from two other technology providers early on who were unable to continue to participate as the pilot got started.

Perhaps the most significant impact of these challenges was that we were unable to complete our initial plan of performing both an initial iteration of RABET-V and a subsequent product revision iteration. Instead, we were only able to complete the full initial iteration for each of the three products. Performing a full initial iteration allowed us to completely define and test the RABET-V Program Description, evaluate the effort required for testing, obtain actual process and product maturity scores, obtain preliminary cost estimates, and evaluate the risk-metrics produced. However, by not completing a follow-up product revision iteration, we are unable to make definitive claims about how the risk-based approach will streamline product updates, the time those updates will take, and the cost of those updates. Gathering that necessary information will have to be done in subsequent RABET-V efforts.

# 3 Research Questions and Findings

In this section, we discuss our findings based on the pilot program research questions. These questions were developed prior to the start of the pilot to provide a more objective basis for assessing the pilot's outcome.

To reach the findings in this section, we interviewed each technology provider, the steering committee, and each internal resource that took part in developing and administering the program. Where relevant to the context of the response, we identify the role played by the respondent but do not identify the individual or organization. In cases where the response may be sensitive, we omit the role to prevent disclosure of the respondent's identity. Most of the questions have qualitative responses, though some are informed by time and cost tracking to provide a more concrete view of the resources required to administer and participate in the RABET-V program.

We made changes to the pilot as we learned throughout the process, and this resulted in some of the questions being less meaningful by the end of the pilot. Where that happened, we note why the question is no longer meaningful and discuss the implications of the changes we made.

## 3.1 Time and Cost Implications

- 1 What are the review time implications of the RABET-V approach for:
  - 1.1 The initial verification of a product from a new vendor?
    - 1.1.1 Initial verification of a product from a new vendor requires the full set of services in the RABET-V process, including the process review.
    - 1.1.2 Based on data from the participating technology providers, we anticipate that the time requirements for a technology provider to conduct an initial verification of either an e-pollbook or election night reporting system will require a commitment of 75-150 person-hours of internal resources, with most coming from mid-level resources. This includes all preparatory work, internal meetings, meetings with the Administrator, and follow-up. This does not include the work of developing the product and organization, such as writing internal policies, code, and testing practices.
    - 1.1.3 Based on data from Administrator tracking, we anticipate that the time requirements for the Administrator for an initial verification of either an e-pollbook or election night reporting system will require 60-120 person-hours over a six-to eight-week process, with most coming from mid-level resources.
  - 1.2 The initial verification of a [new] product from a vendor that has previously been through the RABET-V process?
    - 1.2.1 The gain from a technology provider having been through the RABET-V process is that they have a process review complete, even if the product itself is new to the RABET-V process. We estimate the process review accounts for about 25% of the total hours required to complete a verification. Thus, we anticipate 55-115 person-hours to complete an initial verification of a new product by an existing RABET-V technology provider and 45-90 person-hours for the Administrator.

### 1.3 The re-verification of a product?

- 1.3.1 Changes to the pilot resulted in re-verification being moved to the second pilot. We do not currently have data on re-verification.
- 1.3.2 While the extent of testing will depend on the type of change, feedback from technology providers suggests that the testing matrix process will result in efficient re-verification for most changes as the overhead of preparing materials will decline substantially.
- 1.3.3 While we need data from re-verifications to solidify these estimates, based on initial verification data, we believe re-verification will require a single-digit number of person-hours for the technology provider and 8-12 person-hours for the Administrator for smaller changes such as operating system patches over a one-week period. We further estimate about 30-50 person-hours for the technology provider and 30-50 person-hours for the Administrator for larger or more critical changes such as those to security services, over roughly a three-week period.

### 1.4 Other notes

- 1.4.1 In addition to the estimates above, we anticipate efficiency gains through increased experience by the Administrator in conducting verifications as well as additional investment by the Administrator to provide improved materials, videos, templates, etc., to support more efficient verifications. These efficiencies will result in a slight reduction in total time required for verifications on the scale of 10% for both technology providers and the Administrator, though these gains may not be experienced by the technology provider until they have been through at least one re-verification.
- 1.4.2 Contrary to our initial expectations, we believe the two most time-intensive processes—the Process Assessment and Architecture Review—can be incrementally updated in re-verification efforts. We previously thought these processes would have to be conducted in full when there were process or architecture changes. This will have significant advantages in reducing the time and cost of re-verification and will provide additional incentives for continual improvement among technology providers.

## 2 What are the total cost implications of the RABET-V approach for:

### 2.1 The initial verification of a product from a new vendor?

- 2.1.1 Initial verification of a product from a new vendor requires the full set of services in the RABET-V process, including the process review.
  - 2.1.1.1 Based on data from the participating technology providers, we anticipate that the **internal** cost to a technology provider for an initial verification of either an e-pollbook or election night reporting system will be about \$10,000, with most coming from mid-level resources. This includes all preparatory work, internal meetings, meetings with the Administrator, and follow-up. Non-labor costs will be low to negligible.



- 3** Is there a viable economic model for the RABET-V process? If so:
- 3.1** Does it require a government agency to drive the program, similar to voting system certification?
    - 3.1.1** There are several viable economic models, as described in the Operating and Economic Model section of this report. Feedback suggests that some feel a government agency is necessary to drive the program, while others believe that buy-in from government agencies in multiple states would be sufficient to support a non-governmental Administrator.
    - 3.1.2** The phrase “similar to voting system certification” had implications for some stakeholders beyond the basic ownership of the program. The Operating and Economic Model section of this report deals extensively with the different roles associated with executing the RABET-V program, but the upshot is that, even if owned and overseen by the federal government, most of the administration and execution of the program does not need to be run by a government agency.
    - 3.1.3** Carefully consider which roles, if any, are inherently governmental and which can avoid some of the perceived inefficiencies with the voting system process.
  - 3.2** Is there a model that suppliers in the market can support?
    - 3.2.1** The Operating and Economic Model section of this report provides several approaches for a sustainable market for the RABET-V process. See that section for additional details on the points below.
    - 3.2.2** Feedback suggests a range of potential options. The RABET-V process is likely inexpensive enough that many technology providers will be able to bear the expected costs for both their internal activities and the Administrator’s direct costs to verify the technology provider’s products. The model for suppliers, then, may be selected to meet specific market goals, such as supporting the introduction of smaller technology providers to promote a more robust market. The Administrator may adjust pricing schemes such that larger suppliers pay more as a way of subsidizing smaller suppliers.
    - 3.2.3** Maintaining a market may prove less challenging than the initial development of it. The Administrator must simultaneously bring in enough states and technology providers to create value on both sides of the market—i.e., the state side interested in reports on products and the technology provider side interested in successful verifications to show states. The Operating and Economic Model section provides options for effectively seeding the market.
    - 3.2.4** For technology providers, initial options include outside funding (e.g., federal funds, private grants) to seed the Administrator so that those costs do not need to be included in early technology provider submissions and seeding technology providers directly to create incentives for them to go through the RABET-V process.
    - 3.2.5** In the long term the RABET-V process may have a net positive cost impact, in addition to security gains. As vendors improve their organizational processes and architectural maturity to achieve more streamlined verifications, their overall costs may decrease as RABET-V creates incentives for more efficient operations.

### 3.3 Is there a model that states and localities can support?

- 3.3.1 The Operating and Economic Model section of this report provides several approaches to a market sustainable model for the RABET-V process. See that section for additional details on the points below.
- 3.3.2 Feedback suggests a range of potential options, the most likely of which is a model in which a state or locality becomes a Subscriber to gain access to reports. Operationalizing the RABET-V process will need to determine a reasonable cost that balances budget constraints for state and local governments with achieving sustainability of the Subscriber side of the market.
- 3.3.3 Like with technology providers, maintaining a market may be less challenging than the initial development of it. The Administrator must bring in enough states and technology providers simultaneously to create value on both sides of the market. The Operating and Economic Model section provides options for effectively seeding the market.
- 3.3.4 For states and localities, options include initially providing a free Subscriber model and providing funds or other resources (e.g., model program language, agreement templates) to assist states with creating or changing from their current model to one that leverages the RABET-V process.

### 4 Will the process be efficient enough to keep costs low enough for vendors to make minor updates?

- 4.1 Changes to the pilot resulted in CIS being unable to conduct re-verifications. Because of this we do not currently have data on the costs of the various update types, though we have feedback from technology providers having seen the program operate through the pilot.
- 4.2 Feedback strongly supports that the cost will be sufficiently low to support minor updates and that this provides incentive to minimize the number of components changed in any given update and isolate changes within the system as much as possible.
- 4.3 The RABET-V process defines types of changes (e.g., operating system patch, source code change to a security service) rather than size of changes (e.g., *de minimus*, minor, major). It then maps these types of changes to testing procedures. This reduces uncertainty for technology providers and creates more incentive to keep changes small.
- 4.4 There is room to improve. For instance, the RABET-V program could be used to trigger version updates, such as by mapping the software bill of materials for verified products to the National Vulnerability Database, informing both a product's technology provider and Subscribers when a vulnerability is found in an existing verified product.

## Market Maturity Implications

- 1 Is there evidence that products are architected in a manner that is mature enough for the RABET-V process to yield benefits by reducing the extent of re-verification reviews?
  - 1.1 The CIS team hypothesized that there may be a level of architectural maturity below which there is little incentive to improve and above which there are increasing incentives to improve as a competitive advantage.

- 1.2 While a small sample, reviewed architectures show a level of maturity for which we believe maturity scores encourage improvement. In time, this should generate an overall improvement in architectural approaches across the industry.
  - 1.3 During the pilot, we saw evidence that the providers were taking the intermediate results discussed in interviews and using them to improve their processes and systems. This has led to a potential for the Administrator to allow an additional review of the organizational processes at the end of the first iteration, providing an opportunity to improve processes, and thus scores, even before the first verification is complete.
  - 1.4 Will vendors be willing to submit small, frequent updates?
    - 1.4.1 Feedback suggests that technology providers will submit small updates so long as the time required to re-verify is reasonable given the change. A highly streamlined re-verification of small changes will induce frequent submission and strategically timed larger changes.
    - 1.4.2 One challenge may be the extent to which custom deployments constitute a separate release. It would be untenable if each deployment for a technology provider serving many election offices required a separate version for each election office it served; RABET-V versioning must be developed in a way that intelligently allows for multiple configurations while properly testing changes that could impact security.
- 2 Is there evidence that state and local adoption and acceptance processes can leverage the RABET-V process to yield benefits?
- 2.1 Feedback indicated that most states could adopt the RABET-V process without legislative action. There was also a suggestion that even states that do not directly adopt the RABET-V process will benefit: 1) if the state uses a technology provider that did adopt RABET-V and 2) simply by RABET-V setting a standard in the market for a minimally acceptable performance.
  - 2.2 Can states and localities accept RABET-V verifications quickly enough to make the process worthwhile?
    - 2.2.1 Representatives from multiple states indicated that, by leveraging RABET-V verifications, they could make determinations about smaller changes to accept RABET-V verifications and conduct their necessary due diligence without repeating a full testing process of their own. Key to this is that the approach to due diligence also meets the scope of the change. During the feedback session, no states disagreed.
  - 2.3 Will states and localities be willing to adopt new versions at a rate that maintains incentives to put small, more frequent updates through the process?
    - 2.3.1 Representatives from multiple states indicated that with one-time changes to administrative rules, they would be able to make the process more flexible to allow for the types of smaller changes expected through the RABET-V process.
    - 2.3.2 Representatives from the states expressed a desire to make administrative changes in such a way that could allow the RABET-V process to evolve over time without additional changes to the respective states' programs.

## 3.2 Pre-Review Assessment Implications

- 1 Is there a sufficient correlation between process assessment results and verification outcomes to use those assessments to expedite verification and re-verification under RABET-V?
  - 1.1 Changes to the pilot resulted in re-verification being moved to the second pilot. Because of this and the amount of time since the verifications were complete, we currently lack data to make a clear assessment of how well the process results correlate with verification outcomes. That is, we don't yet know how well process assessments correlate with real-world security.
  - 1.2 We did see that process assessment results do not always correlate with Security Service Capability Maturities. An organization that has a less mature process may still have a strong set of security capabilities but will be subject to more rigorous testing for re-verifications. We consider this an acceptable outcome because there are still incentives for the organization to improve its process to speed up re-verifications and reduce their cost.
- 2 Should process assessments be renewed and, if so, how often or under what circumstances?
  - 2.1 We feel confident that process assessments must be renewed but the timing and structure of renewals is unclear. A potential approach is to have a maximum age for a process assessment (e.g., 18 months) but also require that the organization attest to any changes to its process for each product re-verification. These interim changes will be incorporated into the process assessment and reset the clock on the process assessment age. There would also be a maximum time between full process assessments (e.g., 36 months), at which point the organization would be required to undergo another full process assessment.
- 3 Which party is best equipped to conduct process assessments?
  - 3.1 The Administrator should oversee the process assessments, but the execution can be outsourced to experts in organizational process. The Administrator should remain flexible to allow the most qualified assessor for each portion of the RABET-V process. Portions of the process assessment can be self-assessed, but this should only feed into a more thorough, independent process.
  - 3.2 Other models for process assessment may be possible, and this could change the availability of qualified assessors. The Administrator should establish a schedule to review available process assessment approaches and add, remove, or adjust them as appropriate.
- 4 Do architecture reviews provide a sufficient understanding of a given product to determine the impact of *de minimus* system changes? Minor system changes? Major system changes?
  - 4.1 We initially believed that we would establish this three-tiered approach to system changes. As the pilot progressed, we realized that we could instead create a more specific set of system change types and map those types to a testing matrix in which smaller, more targeted changes result in more streamlined testing.
  - 4.2 Existing tool-based methods can analyze software for changes to make this identification a quick and accurate process.



- 4.3 This increases incentives for technology providers to submit smaller changes for re-verification and to improve process and architectural maturity scores.
- 5 Should architecture reviews be renewed and, if so, how often or under what circumstances?
  - 5.1 Re-verifications should include an attestation to whether any architectural changes were made. Based on the nature of the change, the Administrator can decide the extent of an architecture review, if any.
  - 5.2 In addition, information from submissions and the Administrator's analysis tools should identify anything beyond a basic code change. This can serve as a backstop to the attestations.
  - 5.3 Finally, there should be an age-based component to architecture reviews. This can follow a similar approach to that of the process review where there is a maximum age, interim reviews that can reset the clock, and a total time allowed between full reviews.
- 6 Which party is best equipped to conduct architecture reviews?
  - 6.1 The Administrator should oversee the architecture reviews, but the execution can be outsourced to experts in IT architecture. Portions of the process assessment can be self-assessed, but this should only feed into a more thorough, independent process.

### 3.3 Technical Evaluation Implications

- 1 For which types of non-voting election technology will the RABET-V process work?
  - 1.1 Feedback from stakeholder groups confirms that the RABET-V process can apply to virtually any type of non-voting election equipment, though some details of the process may vary. Generally, the RABET-V process is a software verification process and, like software itself, can be adjusted to suit a wide variety of purposes, even beyond elections.
- 2 Is it better suited for some types of technology over others?
  - 2.1 For components built largely on COTS hardware with minimal customization, which includes nearly all non-voting equipment, the process should perform equally well. This includes components from physical deployments, such as e-pollbooks to pure cloud services like many election night reporting systems.
  - 2.2 The RABET-V process is, however, software focused. For components for which customized hardware plays a significant role, it would need significant adjustments to certain sets of requirements, such as boundary protection. Such a case would likely also need to include additional reviews for hardware security. The flexibility of the RABET-V process would support this, but there would be more overhead in establishing this verification approach.
- 3 How, if at all, does the process have to be modified to make it more suitable for different types of non-voting election technology?

- 3.1 The technical requirement sets for different components will vary. It is important to recognize that the technical requirements differ greatly from business requirements. To a non-technologist, the various non-voting components appear to be very different from each other. To a technologist, they are a rearranging of well-known technologies to meet different business requirements.
  - 3.2 The security needs differ as some components will be physical and others cloud-based, some internet-facing and others simply internet connected. This means that the set of applicable security requirements will change for each election component.
  - 3.3 The process reviews should be unchanged, while the architecture review should have minimal changes between different election components. This variation is accounted for in the architectural maturity rubric that was developed in the pilot. It also allows those vendors that have multiple types of election products to leverage previous runs through the RABET-V process.
- 4 Are vendors more likely to accept the RABET-V process for certain types of equipment?
    - 4.1 Feedback suggests that technology providers are likely to accept the RABET-V process for a wide variety of equipment, though some election components are higher priority than others –see “Other notes” below.
    - 4.2 Both technology providers who participated in the pilot would like to continue to participate in future efforts. Other providers have expressed interest as well.
  - 5 Are states and localities more likely to accept the RABET-V process for certain types of equipment?
    - 5.1 Feedback suggests that states and localities are likely to accept the RABET-V process for a wide variety of equipment, though some election components are higher priority than others.
  - 6 Other notes
    - 6.1 CIS has created a recommended priority order. Stakeholders have suggested that this roughly represents their priorities, which align between technology providers and representatives from states. A rough tentative order is e-pollbooks, election night reporting, voter registration systems, online voter registration, absentee ballot request, electronic ballot delivery, ballot on demand, polling place lookup, and interactive sample ballot.

# 4 Operational and Economic Models

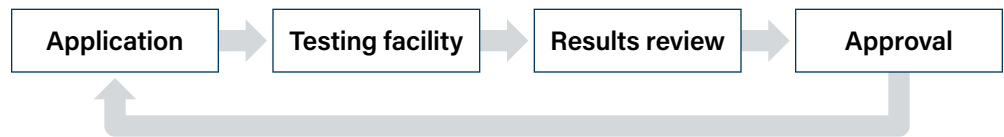
This section describes the roles and processes necessary for a successful, sustainable RABET-V program. The goal of this section is to lay out a broader perspective of how RABET-V fits in the market of state procurement and certification programs. It assumes the RABET-V process exists and describes various approaches to a sustainable market with a RABET-V program playing a central role.

## 4.1 The Current Environment

Today, a variety of models exist within states for certifying non-voting equipment.

- In Ohio, testing of e-pollbooks and electronic ballot delivery systems requires a provider to use an independent testing authority to certify that the state's requirements are met. If successful, an application with a nominal fee and a variety of documentation must be submitted to the state, which is reviewed by the Ohio Board of Voting Machine Examiners. The examiners make a recommendation to the Secretary of State, who approves a certification. Changes to the systems may require this process to repeat in its entirety.
- In California, testing of e-pollbooks and remote accessible vote by mail systems (i.e., electronic ballot delivery and marking systems) takes a different path to achieve very similar results. Applicants must open an escrow fund to cover testing costs along with other required materials. An RFP may need to be issued to test the system, followed by the development of a testing plan. After testing, a report is developed, a public hearing occurs, and the Secretary of State makes a final determination.
- Indiana requires similar testing of e-pollbooks. After submission of an application and certification checklist, the oversight program directs the provider to choose an approved testing lab for functional, telecommunications, compliance, and other testing. If successful, the vendor must conduct a field test. If successful in all phases, the Secretary of State makes the final certification decision. Changes to the e-pollbook may require this process to repeat in its entirety.

Each of these processes has unique features, but they all follow a basic pattern:



While these models can be successful in any given state, they have two major deficiencies, both of which have substantial economic and security consequences. The first deals with each process itself, and the second deals with the implications of having wholly separate processes in each state.

First, over time, the recursive approach to administering these programs—requiring a full review with changes—is increasingly at odds with a modern software and systems development process that relies on smaller, more frequent changes and fewer major updates. Moreover, software and platforms are being provided as a service, complicating the testing approaches.

For hardware, this is generally manageable within the approach of states like Ohio, California, and Indiana. For software, it creates a dilemma: determining when to perform a complete retest and when to allow an update without testing. Testing too often is costly while testing too infrequently will allow vulnerabilities to persist on deployed systems.

Even if these issues are addressed sufficiently within the confines of each state program, a second problem exists with scaling nationally. Technology providers commonly serve more than one state with the same equipment, meaning that for each certification process they currently require separate testing and incur testing and verification costs multiple times. Additionally, the states must develop, publish, and update their own programs, a costly endeavor that underscores why some states have no program at all. This raises costs to both the states and the technology providers, costs that ultimately get passed on to taxpayers.

## 4.2 A Better Way Forward

Developing a more effective approach is possible. Keeping it compatible with the needs and expectations of states is also possible, though requires more careful planning. Achieving both would require two critical changes:

- 1 Establishing a sufficiently flexible process to meet the needs of modern development by testing only what is needed when needed based on the actual changes to the technology.
- 2 Increasing reusability across state programs of as many aspects of the certification process as possible, while still respecting the sovereignty of states and the need to adjust for differences in local requirements.

There is enough commonality across state programs to design a model that achieves, to a high degree of completeness, the first change. This is the RABET-V process we piloted. The RABET-V program takes a risk-based approach to verify product revisions, where the risk estimate is based heavily on the product architecture and the provider's software development processes. Other sections of this report can provide insight into the RABET-V approach.

Achieving the second change requires a new operating and economic model. The remainder of this section will describe the roles and processes of such an environment and provide several options for models that may prove successful. While states and localities will remain the certifier, RABET-V endeavors to bring as many aspects of testing and verification as possible into a single program that many jurisdictions can rely on.

## 4.3 Roles in a National Verification Model

The RABET-V verification program will involve five required and one optional role to operate at a national level while respecting the sovereignty of the states. They are:

- **Technology Provider:** Submits technology to the RABET-V process with the intent of having that technology adopted by one or more jurisdictions.
- **Subscriber:** A state or local government entity that leverages the RABET-V program. Subscribing enables the entity to access verification results and related information.

- **Approval Authority:** A state or local government entity that makes the final determination on whether a product meets the needs of the adopting jurisdiction. Even within a given state or local jurisdiction, this entity may differ from the Subscriber, such as a review board or locality within a state or its respective Secretary of State office.
- **Administrator:** Responsible for overseeing and executing the RABET-V program. While there is a single Administrator, it may outsource individual RABET-V process steps. Whether it executes all steps or outsources some, we recommend that the Administrator oversees the process and maintains a single agreement with each Technology Provider covering all portions of the process. Regardless of the type of entity, the Administrator must have the trust of both Technology Providers and Approval Authorities if it is to receive, secure, and validate documentation, data, and systems.
- **Testing Provider(s):** Entities responsible for executing certain parts of the RABET-V process. For example, there may be one entity responsible for process reviews and a different entity responsible for performing architecture reviews. Like the Administrator, success of these entities hinges on their having the community's trust.
- **Information Sharing Partner:** While not required to operate the RABET-V program, it may be in the national interest to share information with and receive information from certain entities to improve the overall level of cybersecurity defense. This could include, for instance, the Cybersecurity and Infrastructure Security Agency or the Elections Infrastructure Information Sharing & Analysis Center. This would be accomplished through data sharing agreements and may require aggregation and anonymizing information sent to protect individual Technology Providers.

Some roles can only be executed by certain types of entities:

Role	Viable Entity Types	Explanation
Technology Provider	Any entity that wishes to deploy its technology	This will most often be a private entity, whether for-profit or nonprofit. In some cases, a government may develop its own technology and put it through the RABET-V process, either as an independent verification or to deploy it in other jurisdictions.
Subscriber	State and local government entities	Only state and local government entities with election-related authorities will have a reasonable need to access subscriber materials.
Approval Authority	Government entity	Only state and local government entities could serve as Approval Authorities. While an Approval Authority will likely also be a subscriber, a subscriber is simply a government entity that accesses RABET-V reports. An Approval Authority has the authority to allow a piece of equipment to be used in one or more jurisdictions.

Role	Viable Entity Types	Explanation
Administrator	Federal government or private entity. Private entity could be either for-profit or nonprofit	<p>A federal government agency could serve well as the administrator if it could construct the program in a manner that shields the Administrator from political concerns such as budget cuts, shutdowns, and political interference. This may be possible through a long-term grant mechanism or similar tool, or if the program is able to withstand lapses in appropriations (such as a federal program deemed operationally critical).</p> <p>A private entity, whether for-profit or nonprofit, could fulfill the Administrator role if structured to avoid over-reliance on funding from a single party or set of actors that raises questions of objectivity. One such entity could be a coalition of states established to serve this purpose. Federal, state, and local governments should serve as partners and advisors to administering the RABET-V process, especially with regard to the federal government having access to non-public threat information that could facilitate more effective testing.</p>
Testing Provider(s)	Private, either for-profit or nonprofit	Testing, whether in-house or outsourced, is overseen by the Administrator but is addressed separately because it will need to be accepted by the Approval Authorities as meeting their respective requirements. This may require the testing provider to be accredited as a testing laboratory. More than one testing provider may be used for different parts of the RABET-V process.
Information Sharing Partner	Government and private	Any type of organization could be a partner for information sharing, though the expectations and requirements of that sharing may differ.

#### 4.4 Potential Operating Models

An operating model is the basic process by which an election technology product achieves permission to be deployed in a jurisdiction. Three examples of current state-specific models appear earlier in this section.

This section provides three examples of operating models that could work at a national scale. While the most likely scenario is depicted for each model, the ordering of the steps within each model may vary, giving additional flexibility to the Approval Authority. We expect that all three models will exist; the goal is to evolve an offering that increasingly serves the needs of jurisdictions, reducing one-off programs wherever possible.

Where Approval Authorities believe it necessary to impose additional requirements—security or otherwise—on Technology Providers, they should carefully consider the tradeoffs associated with doing so. Slowing the process can result in less frequent updates, which in turn can slow innovation and degrade system security as threats evolve. To the extent possible, Approval Authorities should work with the Administrator to incorporate needed changes before adding on additional requirements. Where additional requirements are necessary, states should work with the Administrator to sequence them in a way that causes the least friction in the process.

In each of these models, functions of the RABET-V process can be separate entities. For instance, the Administrator might contract with testing labs, penetration testers, or other organizations. They would all feed into the RABET-V process orchestrated by the Administrator.

#### **4.4.1 The Trusted Verifier Model**

In this example, a state or locality reviews the RABET-V process and deems it sufficient as a means of verification. The Approval Authority establishes a threshold for certification based on the various RABET-V outputs.

Under this model, the state may still establish other requirements outside of the RABET-V program, such as meeting specific procurement requirements; usability, accessibility, and functional requirements; or a particular configuration of the product.

For revisions to the product, the Approval Authority would accept a re-verification from the Administrator, which would scale the verification appropriately based on the scope and risk of the change.

#### **4.4.2 The Verifier-Field Test Model**

This example begins with the Trusted Verifier model where the Approval Authority accepts the verification results if they meet a particular threshold.

In addition, if the threshold is met, the product then goes through field testing to ensure that it works under its jurisdiction's real-world conditions. With many types of products, this is more like integration or user acceptance testing.

In this model, the Approval Authority may have additional requirements, just as in the Trusted Verifier model.

For revisions to the product, the Approval Authority would accept a re-verification from RABET-V, which would scale appropriately with the scope of the change. The Approval Authority could specify, based on its own analysis of the change and RABET-V re-verification results, whether field testing is necessary. Ideally the Approval Authority limits field testing to only times when the changes are likely to impact field performance. Requiring field tests when they are unlikely to have different results from a prior iteration risks the gains of the smaller, more frequent update approach.

#### **4.4.3 The Verifier+ Model**

This example again begins with the Trusted Verifier model, but the Approval Authority establishes additional security-related requirements.

Arranging testing for these requirements could follow a variety of approaches, the most efficient of which is to work with the Administrator or Testing Provider directly to establish a supplement for that jurisdiction. The Technology Provider could then opt to have that supplement included in the RABET-V process if it is interested in operating in that jurisdiction.

The remainder of the process could follow the Trusted Verifier approach or the Verifier-Field Test approach, including any additional requirements that fall outside of the RABET-V scope.

For revisions to the product, the Approval Authority would accept a re-verification from RABET-V, which would scale appropriately with the scope of the change. Either the Approval Authority could specify whether a retest of its supplement is necessary, or the Administrator could make that determination based on the scope of the change. Any additional field testing or other requirements would occur at the discretion of the Approval Authority.

Approval Authorities make the determination as to what is sufficient for their jurisdiction and must carefully balance their needs with the realities of modern software development. Those Approval Authorities wishing to impose additional security requirements on Technology Providers should carefully consider the trade-offs associated with doing so. Slowing the process can result in less frequent updates, which in turn can leave systems less secure. To the extent possible, Approval Authorities should work with the Administrator to incorporate needed changes before adding on additional requirements. Where additional requirements are necessary, states should work with the Administrator to sequence them in a way that causes the least disruption to the process.

## 4.5 Potential Economic Models

An economic model is the basic process by which value flows between participants, including money and goods or services. In the economic models currently in operation, the Technology Provider may pay the Approval Authority an application fee, may pay a Testing Provider a testing fee, and may be required to put an estimate of associated costs into escrow. In some cases, the testing fees can be substantial, creating disincentives to innovate and stay current with the threat environment.

This section provides several potential economic models that leverage the RABET-V program to scale testing. If properly executed, an initial verification should be no more expensive than it is today—hopefully, less expensive due to incentives for increased up-front focus on security on the part of technology providers—and re-verifications should be substantially faster and cheaper depending on the scope of the submitted changes.

Any model is likely to incorporate multiple approaches below, effectively a hybrid of several options. But understanding the implications and trade-offs associated with each will help establish an appropriate balance and long-term success.

### 4.5.1 Technology Provider High-Subscriber Low Model

In this model, the Technology Provider would bear most of the cost of verification as well as some of the Administrator's overhead. The Technology Provider would pay up-front to begin the RABET-V process, covering the full cost to assess its organization's process. The Technology Provider would then pay its full costs for the initial verification of each product. Initial verifications would take longer and be more expensive than subsequent re-verifications. With less extensive changes in any given re-verification, that review would be faster and cheaper.



Subscribers would have a small subscription cost that gives them access to reports for products that have gone through the RABET-V process. This approach increases the likelihood of bringing in a larger share of subscribers more quickly. Subscription costs could be per report accessed or periodic for access to all reports.

This model has dynamic risks over time as the number of initial verifications will fall over time in favor of updates. This could make it more difficult to model and manage the Administrator's overhead cost. Additionally, because the cost is driven by an analysis of the changes, neither the Technology Provider nor the Administrator would know the exact cost of the verification until partway through the process, though over time the Administrator should get good at estimating and automation should increase consistency. An expected low-cost revision could become much more expensive if a review identifies a need for more expensive parts of verification, such as penetration testing. Technology Providers especially may balk at this.

#### **4.5.2 Technology Provider Low-Subscriber High Model**

In this model, costs are minimized to the Technology Provider to stimulate a more robust set of initial verifications. Likely this would mean using Subscriber funds to subsidize initial verifications, while Technology Providers would pay full freight for re-verifications. Subscriber costs would cover Administrator overhead.

Subscription costs could be per report accessed or periodic for access to all reports.

This model has a much more predictable revenue stream than the technology provider high-subscriber low model. It can encourage a more robust Technology Provider market as today's Technology Providers would face similar or smaller financial burdens while more new, smaller players would be more likely to be able to afford the initial verification.

By placing more of the Administrator's revenue on Subscribers, it would also reduce the risk of undue influence from large Technology Providers.

#### **4.5.3 Technology Provider Subsidization Model**

To draw more small players into the market, this model adds a revenue or market share component to the charge for any Technology Provider. Smaller players would receive a subsidy to promote a robust market and innovation.

To avoid any impression of market favoritism, the rate of charges or subsidization should be predetermined based on revenue, market share, or a similar metric for transparency.

While this model has significant potential to bring in new players and innovation, large Technology Providers would provide such a large share of the Administrator's revenue that it could create the risk, real or perceived, of undue influence by those large Technology Providers.

#### **4.5.4 Government Technology Provider Subsidization Model**

Many governments have their own home-grown solutions but highly constrained resources. To increase public confidence in more of these solutions, this model provides a subsidy to governments for submitting their own systems.

This could also help generate more novel solutions, fit-to-purpose solutions for jurisdictions with unique needs, and, for successful home-grown solutions, more opportunities for adoption within other jurisdictions.

#### 4.5.5 Technology Provider Listing Fee

Part of the value of the RABET-V program is that once a Technology Provider has an established process with the Administrator, it can complete an initial verification for any number of products more quickly and inexpensively. This benefits all Technology Providers but has disproportionate benefits for larger Technology Providers that have more products.

Because this could be an ongoing benefit to the Technology Provider for multiple years, the Administrator may want to smooth revenue by charging the Technology Provider an annual listing fee to have the Technology Provider's reports available to Subscribers. This will also provide Technology Providers with increased incentive to put more products and more frequent updates through RABET-V.

A listing fee would create a hurdle for any given Technology Provider's first product, but once they are part of the RABET-V process, they have stronger incentives to stay in it.

### 4.6 Overcoming Early Economic Barriers

Large-scale models like those described in this section often struggle to gain traction because they are two-sided markets that exhibit positive network effects. More plainly, their value to each type of party (Technology Providers and Subscribers) only grows large when there is wide acceptance by the other type of party. If all the states and territories bought into the model immediately, a Technology Provider would know that undertaking the RABET-V process would have wide acceptance and result in a large market for its products. Similarly, if all Technology Providers had gone through the process, states and localities would have a large incentive to become Subscribers. The challenge, then, is to overcome this chicken-and-egg problem. Several approaches can help do that, and they can be used in conjunction with each other. The remainder of this section provides options overcoming these startup challenges. Each of these will require resources, and the Administrator will have to determine where to get those, such as from government funding, foundations and philanthropy groups, private investment, or other sources.

#### 4.6.1 Seeding Administrator Functions

An initial funding stream to cover the early costs of the Administrator functions would give the Administrator an opportunity to run at a reasonable operating capacity and focus on marketing and recruitment to build to a viable and self-sustained steady state.

#### 4.6.2 Seeding Technology Providers

To overcome the two-sided market problem, seed funding for an initial set of Technology Providers could generate an early rush into the market, and then a reputational effect in which other Technology Providers see that being verified through the RABET-V program is becoming a standard practice.

Not only would this get Technology Providers into the market, but the relatively low incremental cost of revisions would help establish a healthy re-verification process early. In addition, it would help the Administrator refine its processes to lower costs and speed verifications.

#### **4.6.3 Seeding Approval Authority Changeovers**

A source of friction for establishing the new operating model will be Approval Authorities establishing new policies to accept RABET-V verifications in lieu of current processes. While this change will be easy and inexpensive for some states, others may require more extensive efforts. Resources to conduct these changeovers could solve the acceptance problem, generating incentives for Technology Providers to gain verification.

This could involve providing funding to numerous Approval Authorities, but also by documenting early case studies and templates that other Approval Authorities could use to hasten the process and reduce the resource required by any individual Approval Authority.

#### **4.6.4 Offering the Subscriber Side of the Market for Free**

Until there is a robust set of Technology Providers through the RABET-V program, Approval Authorities may be reluctant to pay to be a Subscriber. One of the most common ways to overcome this is to allow Subscribers onto the platform free of charge. This could be temporary based on time or a fixed number of reports, after which a Subscriber fee begins. This could draw in Subscribers, inducing Technology Providers to enter the other side of the market.

### **4.7 Conclusion**

While there are many pieces to both the operating and economic models, the conditions exist today to establish a stronger, more responsive market for non-voting election technology. By coupling verification process changes already underway in RABET-V with an operating and economic model that can ultimately reduce costs to the states, a stronger national model can provide efficiency and security benefits without sacrificing the sovereign roles of states in administering elections.

# 5 Recommendations

We believe the RABET-V pilot was a success, especially given its execution during the difficult 2020 election season.

That said, there is a long way to go to making the RABET-V process a viable, sustainable operation in the U.S. election community. This section lays out recommendations for improving the current process, establishing an operational structure, and enhancing the service offerings in the future to provide more value to technology providers and election offices.

The recommendations are organized into categories below.

## 5.1 Preparation

The pilot served as both a trial run of the RABET-V process and as part of the design process. Because of this approach, we had a limited set of material that Technology Providers could use to prepare their submissions. This was a major inhibitor of efficiency for both the Technology Provider and the Administrator.

To overcome this, the Administrator should create an addendum to the Program Manual or a separate set of documents that provide reference architectures, sample diagrams, and other examples of a quality initial submission. This can evolve over time and should include an annotated version of the application, showing common sticking points, notes and clarifications. It should also have more information about the submission process to encourage Technology Providers to have the appropriate team members ready to engage at the right times, especially for the Process Review.

## 5.2 Iterations

The challenges of 2020 led to an abbreviated pilot in which we were unable to test re-verifications. While we have high confidence in the time and cost efficiencies of the developed methodologies, as this is a crucial aspect of the RABET-V value proposition, we recommend additional testing of the RABET-V streamlining process to ensure the proposed approach to re-verification meets expectations for security outcomes. That is, a rigorous effort should be made to validate the assumption that testing can be streamlined for smaller changes.

## 5.3 Process Steps

### 5.3.1 Human vs. Tool-Based Reviews

During the pilot, we deployed a mix of people and tools to conduct separate parts of the RABET-V process. This ranged from a fully-manual approach for the Process Review and a highly automated approach for the Architecture Review via a tool called Lattix. This tool proved valuable both as a system analysis tool and a change analysis tool. Using such tools can hasten verifications but comes at a cost. Having a more concrete sense of the RABET-V process in action, the Administrator should conduct a thorough study of which combination of manually and automated processes achieves the desired security outcomes at the desired pace and price.

### 5.3.2 Incremental vs. Full Reviews

As mentioned earlier in this report, we assumed we would need to conduct full Process and Architecture Reviews when changes were made. During the course of the pilot, we determined that RABET-V could support a more agile approach for both reviews. As part of a re-verification, the Technology Provider could submit changes to its processes or architecture for review, have them assessed, and receive an updated process score.

An improved score could create additionally streamlined testing for software changes. On the other hand, should a Technology Provider take on riskier organizational processes, it could increase the testing burden on the software. This more agile approach is highly consistent with the goals of RABET-V, and we strongly recommend integrating iterative process and architecture reviews as part of the standard RABET-V process, while maintaining full reviews over time. This approach is also consistent with many standards' certification processes which, for instance, require yearly reviews and a full assessment every three years.

### 5.3.3 Combining Security Claims Validation with Verification Testing

During the pilot we saw that some steps were sufficiently independent that they could occur in parallel, while others, particularly the Security Claims Validation and the Verification Testing, were so intertwined they truly only made sense to occur together. We recommend combining these two functions into a single process step that ultimately generates the RABET-V testing results.

## 5.4 Automation

While the RABET-V process provides for efficient reviews of small changes and much more closely aligns with the modern software development model than traditional testing regimes, there is room to grow.

We recommend the Administrator explore the possibility of integrating tools directly within Technology Providers' continuous integration and continuous deployment (CI/CD) pipelines. A fully integrated testing regime can maintain the independence of the Administrator from the system but provide automated verification within a Technology Provider's deployment model. A fully integrated RABET-V process within CI/CD would allow verifications triggered by the deployments themselves, with intervention by the Administrator only when triggered by the scope of the change itself. These benefits come with costs of integration setup, maintenance, and additional security reviews and controls; these complex trade-offs require careful investigation.

## 5.5 Economic Model

### 5.5.1 Starting Up

Finding a sustainable, steady state cost model for the Administrator is important, but will be less of a challenge than starting up an economic model that doesn't currently exist. To that end, we have a few important recommendations. Like everything in this recommendations section, this section reflects the opinions of the CIS team that worked on the RABET-V pilot and should be taken only as one perspective of many.

- 1 Focus first on Subscribers. While fostering a robust Technology Provider market is also critical, by eliminating duplication of documentation across multiple state certification programs, the RABET-V process can already lower costs to Technology Providers which, along with a strong initial subset of states adopting RABET-V, should provide sufficient participation from Technology Providers.
- 2 Choose election components wisely. Put simply, there is a larger market and more money in some election technology components than others. To be successful, the Administrator should conduct a phased rollout of supported election components beginning with those that have the greatest likelihood of fostering a sustainable cost model.
- 3 Don't over-complicate. While the internal reviews and testing regimes may require a relatively high degree of sophistication, these should be packaged in a way that allows less sophisticated Subscribers to make strong, defensible decisions. Providing significant detail to Subscribers will support Subscribers with better technical resources, but clear and concise data on how a product performs will support clear justifications in the highly political environment of elections.
- 4 Support state transitions. In addition to keeping Subscribers' costs down, we recommend significant investment in helping states either develop or transition to a state certification program that supports RABET-V. This could mean the Administrator providing model legislation and regulation language, providing assistance in crafting program procedures, providing training to states and localities on how to interpret results, and other similar efforts.

### 5.5.2 Administrator Duties

The RABET-V pilot took a distributed approach to administering the program. While done for convenience in the pilot as it would be impractical to staff up, the pilot demonstrated that such an approach may not only be possible, but preferred.

Just as a healthy market for election technology products can result in better security outcomes, a diversity of process providers within the RABET-V process can result in better security outcomes. To that end, both specialization in steps of the RABET-V process as well as having multiple providers for each of those steps can result in better security outcomes and more agility for the Administrator.

Moreover, this allows the Administrator to focus on innovation and efficiency and to avoid finding itself entrenched in its own approaches. The security goal of modern software development is to match pace with or outpace adversaries. This creates a need for an Administrator that can accept innovation from Technology Providers and adapt its practices without sacrificing speed or security. This may be possible with a well-run integrated Administrator shop but is more likely if the Administrator is able to quickly seek new assessment and testing providers and keep current providers in a position to adapt to changing needs.

## 5.6 A Final Word

This process began with a widely-held view that many existing testing regimes are unable to keep pace with the rapid deployment model in practice today. We hypothesized that we could develop a new model that better addresses today's threat environment.

Throughout piloting the RABET-V process we learned a great deal about what did and didn't work and what could generate even greater security gains for the election community. We have strong confidence that the RABET-V process can improve security outcomes at a lower cost with greater speed and flexibility than traditional testing approaches.

That said, the specific approaches matter. The technical and procedural features that can test software efficiently are only part of the equation. The implementation of process and architectural reviews is what makes the streamlined testing possible. The Administrator must doggedly adhere to the principles of the RABET-V process's holistic approach to enable an abbreviated approach with any given re-verification. Failing to do so presents the dangerous proposition of a process that adjusts to accelerate verifications but fails to properly account for risk in those adjustments.

Additionally, continual improvement of the RABET-V process itself is what will allow it to remain an innovative and effective verification regime into the future. The value of the RABET-V process is not its newness, but is its focus on continual renewal—renewal of both the products it seeks to verify and its own internal approach.




The Center for Internet Security, Inc. (CIS®) makes the connected world a safer place for people, businesses, and governments through our core competencies of collaboration and innovation. We are a community-driven nonprofit, responsible for the CIS Critical Security Controls® and CIS Benchmarks™, globally recognized best practices for securing IT systems and data. We lead a global community of IT professionals to continuously evolve these standards and provide products and services to proactively safeguard against emerging threats. Our CIS Hardened Images® provide secure, on-demand, scalable computing environments in the cloud.

CIS is home to the Multi-State Information Sharing and Analysis Center® (MS-ISAC®), the trusted resource for cyber threat prevention, protection, response, and recovery for U.S. State, Local, Tribal, and Territorial government entities, and the Elections Infrastructure Information Sharing and Analysis Center® (EI-ISAC®), which supports the rapidly changing cybersecurity needs of U.S. election offices. To learn more, visit [CISecurity.org](https://CISecurity.org) or follow us on Twitter: @CISecurity.

 [cisecurity.org](https://cisecurity.org)

 [info@cisecurity.org](mailto:info@cisecurity.org)

 518-266-3460

 Center for Internet Security

 @CISecurity

 TheCISecurity

 cisecurity