Bridging the Cloud Trust Gap: Using ORCON Policy to Manage Consumer Trust Between Different Clouds

Stephen S Kirkman\textsuperscript{1} and Richard Newman\textsuperscript{1}

\textsuperscript{1}Computer and Information Science and Engineering, University of Florida, Gainesville, FL, 32611 USA
kirkman@ufl.edu, nemo@cise.ufl.edu

Abstract—If we send our data to the cloud, how do we know the cloud will not outsource our data to another cloud provider? We leverage the originator controlled access control policy (ORCON) which is originator focused and provides the opportunity to express consumer specific desires relating to our data in the cloud. We present a novel cloud trust policy framework which allows for the consumer to make more expressive choices regarding the handling of her data across distributed clouds using ORCON as a mechanism. We also identify a trust gap between a cloud consumer and the cloud. A cloud consumer (specifically first time user), who is provided with privacy and security guarantees and promised a level of service, must decide to take a leap of faith and trust the cloud. This ‘trust gap’ is a both short term and long term trust issue. We explore ways to bridge this gap. Finally, we identify a conflict-of-interest that exists within the cloud trust research space and explain how to mitigate it. This is achieved principally by moving attestation servers out of the cloud that it is supposed to be attesting to. This effectively creates Attestation as a Service (AaaS).

Keywords—Cloud Trust, Cloud Security, Distributed Clouds, Fog Computing, Attestation as a Service, ORCON

I. INTRODUCTION

The cloud has seen an explosion of attention in both academic research and industry, but there is still widespread concern over the lack of trust.\cite{1} There are still challenging problems to be addressed in the cloud trust research space. There is an emerging research area in cloud computing called Inter-cloud VM migration\cite{2, 3} which significantly impacts cloud trust. It involves the movement of data between different clouds and possibly across geo-political boundaries.

There is a gap in trust when a new cloud user first starts using the cloud. As a new consumer uses the cloud, there is an increase (or decrease) in trust in both the underlying cloud technology and the cloud provider. Many cloud customers accept the security guarantees of the cloud at face value and begin to use the cloud. Regardless of these security assurances and guarantees, the lack of direct experience with that cloud provider negatively impacts trust until the consumer has had positive first-hand experience. We provide approaches to fill this gap with the objective of alleviating this lack of trust. This also encourages overall trust in cloud computing technologies.

Organizational trust and interpersonal trust are well researched topics. The notion of trusted computing entered the IT vernacular with the formation of the Trusted Computing Group (TCG) in 2003.\cite{4} The TCG supports development of the Trusted Platform Module (TPM).\cite{5} The TPM provides for trusted computing via integrity guarantees and a trusted boot process. The TPM is used as the basis of many different research proposals in cloud trust \cite{6–10}. Other research projects track trust level based on service metrics or real time data to support the consumer decision process.\cite{11–13}

When we give our data to the cloud, will we know that our data has been migrated to a different cloud provider possibly across geo-political boundaries? In today’s cloud, one organization can have multiple data centers or be part of cloud federations each organization having different policies. Our data and virtual machines might be spontaneously migrated beyond boundaries. How can consumers request this not happen in advance? The SLA is designed for one provider/consumer relationship, but SLAs leaves no expression for consumer specific desires, certainly not multiple clouds. We focus on providing a mechanism for the consumer to articulate these and other data requirements.

There are few mechanisms/protocols in use (some in research) that will allow us to specify how data can be passed between clouds. We use the Originator Control (ORCON) access control policy as a driving factor behind our proposed framework. ORCON is about data access and control. We review the applicability of common access control policies to this problem.

We also suggest that many current cloud trust architectures \cite{6, 9, 10, 14–16} contain an inherent conflict-of-interest. Some even go further by assuming that attestation functionality in the cloud is assumed to be trusted.

Our target framework addresses these specific goals:

1) Provide for data policies across different organizational boundaries (between clouds using ORCON access control policy)

2) Provide an attestation architecture to confirm data movement between clouds without a conflict-of-interest

ORCON provides us the opportunity to tell the cloud provider via policy which clouds (organizational/political) we do and do not trust with our data. The ORCON policy is an originator access policy. The ORCON policy has been used in the intelligence community as a control specification. We use it as a design philosophy for a framework that ensures the
creator can express her desires and retain control of her data in a distributed environment.

An attestation or consolidation server that is located within the cloud is vulnerable to manipulation from employees of that same cloud. We minimize this conflict of interest by moving the attestation server out of the cloud organization to the control of a trusted third party.

The main contributions presented in this research are:

1) Illustrate the ‘trust gap’ and ways to fill it to enhance consumer trust
2) Provide methods to minimize the conflict-of-interest while collecting/processing attestation data
3) Propose a distributed cloud trust policy framework using principles of an ORCON access control model. This model has two parts:
   a) Data policies between organizational boundaries (between clouds)
   b) Attestations for data movement between clouds, virtual machine co-location, geolocation

The rest of this paper is organized as follows. Section II discusses existing research in cloud trust. Section III discusses how we trust the cloud and the notion of Trusted Computing. In Section IV, we describe the consumer trust gap and how to bridge it. Section V and VI discuss ORCON and our distributed policy framework. Section VII describes future work and section VIII is our conclusion.

II. EXISTING RESEARCH IN CLOUD TRUST

Some existing research in the cloud trust research space include introducing middleman organization, labeling data at the system level within a single cloud and systems to prevent malicious feedback. The descriptions, pros, and cons are summarized in the Table 1.

III. ELEMENTS OF TRUST IN A CLOUD

Many systems in the cloud trust research space use hardware based roots of trust for integrity attestations. There are other methods to support trust in cloud computing. Data provenance attempts to capture everything that has happened to data from its inception, which requires a persistent repository for provenance storage. Homomorphic encryption provides confidentiality. It allows the cloud to perform computations on your data without revealing the data to the cloud. Fog computing is a current trend that might encourage cloud trust. It is an effort to extend the cloud closer to the edge where mobile users reside.

A. Trusted Computing

When accessing the cloud, the consumer no longer has direct control of their data. Industry and research have provided a starting point for trust. The computing paradigm refers to applications and systems that “leverage hardware-based roots of trust”[5] a.k.a Trusted Platform Modules (TPMs) or Software Guard Extensions (SGX). The TPM is a chip that is already present in commodity hardware. Intel makes SGX, which is a set of Instruction Set Architecture (ISA) software extensions that can create a trusted enclave. While there are many research projects that use the TPM [6–10], it has limitations. It provides integrity checking only during the boot process. Other research has extended this to include runtime integrity for limited cases.[20] In most cases, run-time security must be left up to other applications.[5, 21, 22] The TPM provides assurance that we know the software we are booting is the software we expect to be booting. However,

<table>
<thead>
<tr>
<th>System</th>
<th>Description</th>
<th>Pros/Cons</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cloud Trust Protocol [17]</td>
<td>Proposes unified standardized API, allows customers to query the cloud for info related to various service attributes</td>
<td>Pros: Fully developed API. Cons: Cloud has the option to respond (or not) to customer requests for current service data, does not cover the actual monitoring infrastructure.</td>
</tr>
<tr>
<td>Trust Model [13]</td>
<td>A formal trust model for SaaS which presents a quantitative approach to model trust by defining different space-based and time-based trust ‘properties’</td>
<td>Pros: Attempt to Quantify Trust. Cons: SLA is used as the basis of trustworthiness and the trust model is calculated after the determination is made.</td>
</tr>
<tr>
<td>Cloud Access Security Brokers (CASB) [18]</td>
<td>Security policy enforcement points in between cloud and consumers. Security middleman, consolidating security policy, sign-on, and more</td>
<td>Pros: It takes trust role out of cloud. Cons: This is a centralized model and the brokers would need to be vetted via a trusted third party. It shifts trust.</td>
</tr>
<tr>
<td>Policy-Sealed Data [9]</td>
<td>Implements a ‘policy-sealed’ data trust system for a single cloud domain. It seals (encrypts) and unseals (decrypts) customer data based on the trustworthiness of a cloud node.</td>
<td>Pros: They take steps to prevent some internal mismanagement threats. Cons: System relies on an internal monitor. To ensure the authenticity and integrity of the monitor, “customers must directly attest the monitor when first using the system”[9] This process is not described.</td>
</tr>
<tr>
<td>Data Location Control Model [19]</td>
<td>Model provides users visibility into location of data and are notified when location changes using XML based policies.</td>
<td>Pros: We believe there is a need for more consumer choice and visibility into data location/migration. This model works toward that. Cons: Policies are stored within the cloud. When migrating across organizations, policy store must be copied to new organization. We choose to integrate them with the VM, so there is less overhead.</td>
</tr>
</tbody>
</table>
the TPM cannot address run-time protections from malware installed or reconfigured after boot time or malicious co-resident virtual machines.

B. Transparency - Data Provenance

Many cloud security researchers believe that transparency is the key to trust. Publicly traded companies have some transparency through the Securities and Exchange Commission. They must follow rigorous financial and compensation disclosure processes which, in some cases, includes business operations.[23, 24]

The cloud offers little transparency because of the nature of the Internet and virtual machine technology. How much transparency we demand from the cloud is dependent on the importance of the data. Metrics like uptime or availability of data are very transparent. However, detecting a loss of confidentiality is not transparent at all. Since data provenance focuses on recording actions performed on data from its creation onward, this record can be used to prove when data has been copied, moved or been accessed. Provenance data, cannot prevent data from going places it shouldn’t go. However, this record can support cloud trust by verifying what we expect to happen and potentially verifying what we expect not to happen. This makes security and protection of provenance data all the more vital.

C. Homomorphic Encryption

In homomorphic encryption, the data stays encrypted even when the cloud is performing computations. Using homomorphic encryption, the cloud can run a limited number of operations on the cipher text that will mirror the same operations in plain text. Using this encryption technique, the cloud provider can manipulate the data without even seeing the data, thereby assuring confidentiality. But this is limited to what the homomorphic scheme allows. Simple homomorphic encryption usually involves just one operation. In fully homomorphic encryption, instead of just one operation, you have many operations that are supported. You have enough to combine the operations to achieve any program. Fully homomorphic encryption comes with costs: the cloud provider is unable to take actions after the operations, and it is inefficient. The size of the cipher text and the complexity of the encryption grow enormously. It is only good on a small subset of tasks. Even if traffic analysis is performed, information about your computations might be revealed.

Homomorphic encryption is extending into deep learning. Deep learning happens when neural networks meet big data. Deep learning is now working cooperatively with homomorphic encryption in some cases to provide alternatives to cloud confidentiality without needing fully homomorphic encryption.[25]

D. Fog Computing

The goal of fog computing is to extend the cloud closer to mobile users (i.e. fog is closer to end users than clouds). It is an effort to move the processing power of data centers closer to the edge of the network where mobile users reside. This will decrease the latency of mobile users’ computations. It also could have a positive impact on cloud trust if these edge servers can be used for both mobile and non-mobile cloud consumers. If edge servers were organized into smaller local data centers - closer to cloud customers - that provided storefront access, the lack of direct control might be mitigated by data being closer. A retail presence with face-to-face interactions goes a long way to foster trust.[26]

IV. BRIDGING THE CONSUMER TRUST GAP

After a consumer has assessed the guarantees provided her, there still exists a ‘trust gap’ because trust is not instantaneous. This gap in trust starts when the consumer first uses the cloud. As a new consumer uses the cloud, there is an increase (or decrease) in trust in both the underlying cloud technology and the cloud provider. Some cloud customers might just accept the security guarantees or reputation of the cloud at face value. Regardless of these security assurances and guarantees, the lack of direct experience with that cloud provider negatively impacts trust until the consumer has had positive first-hand experience.

Filling this gap serves several purposes. First, it gives the consumer extra confidence or benefit in the early stage of using a cloud. Second, it provides an additional mechanism to support trust for new or newly established cloud providers who do not have reputation. Bridging this gap might attract even more consumers who had previously decided not to take the leap of faith. Lastly, many times, only those who are proficient in security will be able to interpret the security aspects of the SLAs. Consumers might not be interested in or have the time to understand the technical details behind the security protections. To fill the gap, a cloud must provide tangible incentives. We propose the following techniques and enticements:

1) Free cloud insurance: cloud insurance works like any other insurance. Data can be protected at different reimbursement levels depending on the policy. Having insurance can assist both the cloud and the consumer in the process of building long term trust. Companies in this space include cloudInsur[27], MSPAlliance[28].

2) Vouching for a cloud by a Trusted Third Party (TTP). A TTP cannot be associated with the cloud. There is a risk that a long term relationship between a TTP and a cloud provider can interfere with objectivity due to the possibility of collusion. The TTP may be a State, Federal Authority, or any organization certified by an impartial authority.
3) Customer managed approach: the customer puts less critical data in the cloud until trust is built up (i.e., they should test the cloud with lower value data that they can afford to lose.)

We believe what is missing in cloud computing trust is a more customer focused policy. Our proposed ORCON framework would help both short term and long term trust.

V. ORCON BACKGROUND

We suggest a policy framework and protocol that is more expressive and flexible for the consumer based on an ORCON-like policy.

A. ORCON History

ORCON access control is a lesser known access control policy that has been used in the Intelligence Community[29] as a control specification whose focus is on an ‘originator’ controlling the dissemination.

ORCON Requirements at the Organizational Level[30]

Given: \( S \) is a pool of subjects, \( O \) is a set of objects, subject \( s \in S \), object \( o \in O \), then

\( s \) marks \( o \) as ORCON on behalf of organization \( X \) (note: in this instance the organization is the originator). \( X \) allows \( o \) to be disclosed to subjects acting on behalf of organization \( Y \) with the following restrictions:

1) \( o \) cannot be released to subjects acting on behalf of other organizations without \( X \)’s permission; and
2) Any copies of \( o \) must have the same restrictions placed on it.

In our ORCON framework, \( s \) would be the consumer (originator), \( o \) would be the virtual machine, \( X \) and \( Y \) would be cloud organizations.

ORCON versus MAC, DAC, and RBAC

In mandatory access control (MAC), the files are controlled by the system and the labels are assigned by a centralized entity of the system. Owners and users do not have the right to change data labels, nor their own label. In discretionary access control (DAC), the owner can do anything to files they own (including copies that are provided to them); total control is at their discretion. The owner of the file is not necessarily the creator or originator and a user owns any file they copy. RBAC is a role-based system, where control over objects is based on user role. MAC, DAC and RBAC are typically implemented on a per system basis.

Pure DAC fails the ORCON test because the owner is not necessarily the creator and the owner can change permissions at will. Hence MAC is closer to ORCON and could theoretically simulate it by creating a separate category for combination of object/owner/recipient. But the owner does not have control under MAC and MAC does not scale as the categories would explode and it also requires a centralized manager.[30] ORCON is more appropriate than mandatory access control or discretionary access control because ORCON is a control policy that emphasizes “originator’s” desires and it can be used in a distributed context.

B. Related ORCON Research

In 2009, Chen and Lee[22] built a proof-of-concept application that enforces an ORCON-like policy. They leveraged a modified motherboard they developed (i.e., Secret Protection (SP) architecture) [22] and custom software they called a Trusted Software Module (TSM). TSMs adjudicate policies that can be enforced (by the hardware) even if the underlying operating system is compromised. Multiple group encryption/decryption key pairs can be stored in a node (cloud server) hosting the SP device. The TSM will grant access to data based on a policy file and public keys. The proof-of-concept was a replica of the unix application vi.

In 2002, Park et al.[31] suggested licenses and tickets to grant and re-distribute access permissions. Recipients make explicit requests for documents or information. They have reduced the problem of distributed control between trust domains to the need for an implicitly trusted virtual machine on each host. Their virtual machine concept serves as a reference monitor of the trusted computing base. While today’s virtual machines are not used in this manner, the TPM might be able to alleviate some of this problem. However, the TPM does not ensure run-time integrity.

VI. PROPOSED DISTRIBUTED POLICY FRAMEWORK

Classic ORCON involves a single user requesting access from the originator. We use policies that have an ORCON-like mechanism, but tailored for distributed clouds. Our system uses tags that travel with data. These tags express the originator’s policy; we only want the data to go to places that our policy says it can go.

Our framework is composed of five key components:

1) A model to express our policies and the cloud needs to be able to interpret them.
2) A mechanism to attach policies via a tagged architecture and the use of ORCON framework.
3) Cooperation from the cloud provider to willingly commit to agree to and enforce these policies. This could be an addendum to the SLA
4) Positive verification from the cloud in the form of attestations.
5) Negative verification from the cloud; ensure the cloud is not doing what we do not want.
A. Policy Model

Our proposed ORCON policy allows the consumer the chance to express restrictions. In our model, the absence of a specific sub-policy means that the consumer does not wish to be notified of changes directly. However, if either a white list or a black list is provided and a migration happens which is not explicitly listed, the consumer needs to be asked. The following tables represent some of the types of policies that could be expressed via our ORCON policy model:

<table>
<thead>
<tr>
<th>TABLE II GEOLOCATION POLICY MODEL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Policy Entry</td>
</tr>
<tr>
<td>White List Approved</td>
</tr>
<tr>
<td>Black List Disapproved</td>
</tr>
</tbody>
</table>

Note: Geolocation provides information on location of data. These might include geographical requirements (physical distance), network requirements (hops/latency), or geopolitical (crossing political lines).

<table>
<thead>
<tr>
<th>TABLE III VM MIGRATION INTER DOMAIN POLICY MODEL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Policy Entry</td>
</tr>
<tr>
<td>White List Approved Nodes</td>
</tr>
<tr>
<td>Black List Disapproved Nodes</td>
</tr>
</tbody>
</table>

C. ORCON Indirection

In some cases, the consumer’s policy might require indirection in the cloud. The consumer policy might specify approved and disapproved locations or provide a characteristic that applies to class of clouds. The absence of clear direction on an organization would fire an indirection request back to the consumer. Reference figure 1 for the following example.

B. Tag Travels with Virtual Machine

Unlike MAC-type policy where the single organization tells you everything you need to know, we want the originator to set what clouds the consumer trusts. We propose a tag that travels with our data that expresses our policy. This tag will indicate, among other things, in which cloud(s) our data is authorized.

This TAG should be incorporated into the virtual machine when it is created. Relative to the size of a typical virtual machine (order 10 to 100+ Gig)[32], the size of the policy tag is only on the order of kilobytes.

This tag is an XML file that is created with the virtual machine, a suggested example of which is provided here:

```xml
<xml version="1.0" encoding="utf-8"?>
<xml-stylesheet type="text/css"/>
<consumerID="jsmith123">Joe Smith</consumer>
</xml>
</DOCTYPE ORCON POLICY "orcon.dtd">
</POLICIES>
<TITLE>CONSUMER POLICIES</TITLE>
<POLICY>
<ORGWHITE>Org White List</ORGWHITE>
<ORG1>Cloud A</ORG1>
<ORG2>Cloud B</ORG2>
<ORG3>Cloud C</ORG3>
</POLICY>
<POLICY>
<ORGBLACK>Org Black List</ORGBLACK>
<ORG1>Cloud E</ORG1>
</POLICY>
<POLICY>
<ORGBLACK>GeoPolitical Black List</ORGBLACK>
<ORG1>Country E</ORG1>
</POLICY>
<POLICY>
<NUMCOVMS># Co-Resident VMs</NUMCOVMS>
<WHITE>number <= 5</WHITE>
<BLACK>number > 5</BLACK>
</POLICY>
</POLICIES>
```

Note: The knowledge of the owners might fall under privacy rules, however business competition between cloud tenants will want this information

```xml
<xml version="1.0" encoding="utf-8"?>
<xml-stylesheet type="text/css"/>
<consumerID="jsmith123">Joe Smith</consumer>
</xml>
</DOCTYPE ORCON POLICY "orcon.dtd">
</POLICIES>
<TITLE>CONSUMER POLICIES</TITLE>
```
D. ORCON Operations Flow

1) Consumer creates virtual machine
2) Proposed protocol to enter consumers preferences about trusted clouds and other preferences
3) Answers inserted into XML tag
4) Tag is attached to virtual machine
5) VM + tag travel together
6) Cloud uses tag to move data around
7) If cloud needs a policy based answer from consumer, cloud requests feedback from consumer

E. Attestation Architecture

The cloud needs to be able to supply proof that consumer policies are being followed via attestations. Using separate servers within the cloud to manage these attestations are common to the cloud trust research space. Many projects have this architecture: [6, 9, 10, 14–16] and some just assume that attestation management in the cloud is trusted.

To improve cloud trust, there must be some monitoring or data gathering capability that provides for attestations. Attestations support transparency in that attestations can serve as evidence if an event has occurred or a state has been reached. In many cases, the monitoring infrastructure designed into the cloud is planned without consideration of this conflict-of-interest issue. The source of the conflict of interest arises from the potential of organizational goals and employees interfering with the accurate processing and reporting of data. As an example, personal relationships might interfere with job responsibilities in unforeseen ways if attestation administrators fraternize with the cloud team members. This is a fine line, but one that should be addressed.

The greater chance of insiders interfering with the accurate processing and reporting of data. The insider need not even be malicious for there to be a risk. There is a balance between the need to be close enough to the data to gather it, but not so close as to risk manipulation of that data.

External Attestation Capability

We eliminate this conflict-of-interest by moving the attestation collection out of the cloud organization. The entity is external so there is little chance of influence from the target cloud. This also follows separation-of-duties. There still must be an oversight entity who performs inspections and is considered as trustworthy. This could lead to trust a catch-22 (i.e. who monitors the monitor). An alternative is to use a decentralized mechanism. Our proposed attestation server resides in a separate entity and this entity and the attestation server are assumed to be trusted because they must be vouched for by a certificate authority.[33] This effectively creates a service called: Attestation-as-a-Service. See Figure 2.

Attestations

Attestations are the proof or evidence regarding the state of something. Utilizing hardware-based checks via the TPM is an accepted method of making integrity claims about your data. TPM based attestations satisfy integrity at boot time, but should not be the only means we have to verify the state or condition of our data. What is to stop an insider from manipulating attestation information? Nothing can prevent a determined malicious insider. There are other sources of attestation data.

Logs

Logs are not hardware based and are easier to manipulate, but they can provide additional information. In a distributed attestation cloud environment, logs (host based or network based) might help us determine where our data has been. This would be a location attestation. Using logs to attest to the movement of data can encourage checks and balances. If you can reconcile the same activity at two logs or find
inconsistencies, this would work as a good deterrent for malicious intent. In this situation, if a malicious insider altered logs, there would have to be collusion between organizations.

Log Attestation Process

1) Each Node sends its attestation and log data to the attestation server for the consumer.
2) Consumer can request an attestation, or attestation sent at regular intervals, or when an event occurs.
3) Consumer sends attestation request Attest(CloudA, DataTag, Policy) to attestation server.
4) Consumer receives response from attestation server and verifies attestation based on expected information.

As an example of the kind of data to be pulled from network logs, if CloudA sends data to CloudB, then CloudA should have log of the send and CloudB should have a log of the receive. The logs would be distributed, but the chances of them colluding is low. These logs might be also be decentralized, an attestation server might collect logs from these devices, or logs can be reconciled to provide a basis for geo-location attestations.

VII. Future Work

We plan to develop a protocol around our policy framework. We also want to ensure the cloud is not doing what we do not want. This is our fifth tenet. We want to explore various methods of statistical sampling to verify what we do not know. We are also looking into decentralized mechanism such as blockchains.

Sampling of data via a trusted third party might provide a sanity check of the location of data and existence of co-mingling with other data. If the attestation entity can perform random sampling and that random sampling indicates no malicious behavior that would further increase trust in the cloud. Exploring different types of statistics might be meaningful to improve cloud trust. There are no meaningful agreed upon statistics for the perception of trust in clouds.

VIII. Conclusion

There needs to be a solution to alleviate initial lack of trust a new cloud consumer experiences when first using the cloud. We have suggested some techniques to assist the user in increasing their trust and steps the cloud provider can take to help build confidence (e.g. free cloud insurance, vouching by a TTP, using local data storage first). As consumers, we can start using the cloud with low value data. Then, as we have positive experiences, trust the cloud with higher value data.

We discussed the open problem of trusting the cloud. Current research in homomorphic encryption, deep learning, and data provenance also support cloud trust. We discussed the conflict-of-interest that exists when an attestation server is within the cloud and suggested a framework that would mitigate it.

Lastly, we proposed an ORCON distributed cloud framework which is a first step in trust across different clouds. Because of the expanding role of big clouds, little clouds (i.e. cloudlets), and fog computing, the potential for our data to cross organizational boundaries without customer knowledge or permission is greater than ever.

Organizational trust and interpersonal trust have been researched for decades. Trust across clouds is an emerging cloud trust research space. Our framework begins to address this problem by providing an expressive mechanism that allows the consumer more control.

ACKNOWLEDGMENTS

This research was conducted with Government support under and awarded by DoD, Air Force Office of Scientific Research, National Defense Science and Engineering Graduate (NDSEG) Fellowship, 32 CFR 168a.

REFERENCES


