Privacy and Security of Aadhaar: A Computer Science Perspective

Shweta Agrawal Subhashis Banerjee Subodh Sharma
Computer Science and Engineering, IIT Delhi
New Delhi 110016

Abstract

We investigate the privacy and security issues of Aadhaar from a technology point of view. Specifically, we investigate the possibilities of identification and authentication without consent using the Aadhaar number or biometric data, and unlawful access of Aadhaar data in the central repository. Our analysis suggests that privacy protection in Aadhaar will require a) an independent third party that can play the role of an online auditor, b) study of several modern tools and techniques from computer science, and c) strong legal and policy frameworks that can address the specifics of authentication and identification in a modern digital setting.

Index terms: Computer Science, privacy, security, cryptography, authentication, identification

1 Introduction

The Aadhaar project is the world's largest national identity project, launched by government of India, which seeks to collect biometric and demographic data of residents and store these in a centralised database. To date, 1036 million users have enrolled in the system, and the government has spent at least 890 million USD on the project (Wikipedia, 2016a). However, recently there has been considerable deliberations over the privacy and security issues related to the Aadhaar project. In this article, we examine these issues from a Computer Science perspective.

1.1 Background

Privacy concerns relating to the Aadhaar project have been the subject of much heated debate recently (Express News Service, 2016; NDTV, 2016a). On the one hand, positions taken by the government and UIDAI on these issues have been ambiguous. Arguing before a bench in the Supreme Court, the Attorney General of India has claimed that Indian citizens have no constitutional right to privacy (PTI, 2015). This is surprising not only because there are several interpretations of constitutional provisions and judgements to the contrary (Bhatia, 2015; Kumar, 2015), but also because it contravenes conventional wisdom and best practices in digital authentication and authorisation systems (Diffie & Hellman, 1979; Wikipedia, 2016l,g). The finance minister, while getting the Aadhaar bill passed as a money bill, announced that "the government presupposes privacy as a fundamental right" and claimed that the bill has tightened privacy provisions when compared to what was there in the previous version (Scroll Staff, 2016). However, neither the government nor the UIDAI makes it clear what precisely are the privacy concerns that are being addressed, what precisely are the methods being deployed and why the resulting proposal is secure. The UIDAI does

describe the security measures it has put in place (UIDAI, 2014), but does not provide an analysis of the measures with respect to perceived threat levels and potential privacy breaches. This has resulted in an overall confusion about the impact on privacy engendered by the Aadhaar project.

On the other hand, several civil society activists and social commentators (Arun, 2016; Mehta, 2016; Jayaram, 2015; Vombatkere, 2016; Makkar, 2016; Duggal, 2011; Dréze, 2016) have expressed concerns about the weak privacy provisions in the Aadhaar project and bill. However, while alerting to the possibilities of opening doors to mass surveillance, we feel that some of the commentaries have been unbounded in their criticisms and not entirely specific in their statements of concerns. The gist of most criticisms has been that the use of biometrics and a unique identification number, and storage of biometric and demographic data and authentication trails in a central repository, are necessarily unsafe. However, whether breach of privacy is inevitable, and whether there may exist technological and legal provisions which can make Aadhaar safe are important questions that have not been adequately addressed. We note that some crucial lacunae in the identification and authentication processes of Aadhaar have been pointed out in (Centre for Internet & Society, 2016), which also makes several important suggestions including implementation of recommendations of Shah (The Planning Commission: Government of India, 2011) and Sinha (Lok Sabha Secretariat: New Delhi, 2012) committees. Despite these, thorough analyses of the possible ways in which privacy can be breached, and possible countermeasures both from technological and legal perspectives, remain missing. In this note we endeavour to fill in some of this gap from a technology point of view.

1.2 Perspectives on Aadhaar: pros and cons

At its core, the Aadhaar act attempts to create a method for identification of individuals so as to provide services, subsidies and other benefits to the residents of the country. While the effectiveness of Aadhaar to the extent claimed in preventing leakages in social welfare schemes has been questioned (Khera, 2011, 2015; Zhong, 2016), the advantages of computerisation and reliably maintaining eligibility and distribution records in digital forms are well accepted (Masiero, 2015; Khera, 2013). Any digitisation requires indexes or unique ids, and in social welfare schemes local unique ids like ration or job card numbers are typically used. Standardising the digital record keeping processes across geographies and verticals, and linking the local ids with the unique national identities provided by Aadhaar, tantamount to virtually collating the different digital record tables into one. Though the digital records may still be geographically distributed, real-time access to the data, using the Aadhaar ids as handles, can then be provided to authorised central and state agencies for audit, monitoring, analysis and planning purposes. Thus, the Aadhaar number provides a single index across all services that may use Aadhaar.

Additionally, the Aadhaar project may provide the necessary impetus to standardisation and digitisation of other domains as well, many of which are long overdue. The Aadhaar ids can be used to create local ids for digitisation of new verticals easily. Even more importantly, Aadhaar can facilitate linking of local ids in currently isolated verticals like census, education, health-care and immunisation records, birth and death records, land records, property registration, income tax, banking, loans and defaults, police verification and law enforcement, disaster management, security and intelligence and such others. Thus, Aadhaar may not only enable efficient design, delivery, monitoring and evaluation of services in each domain individually, but also offers the possibility of using modern data analytics techniques for finding large scale correlations in user data that may facilitate improved design of social policy strategies and early detection and warning systems for anomalies. For example, it may be tremendously insightful to be able to correlate education levels, family incomes and nutrition across the entire population; or disease spread with income and education. More generally, it may enable carrying out econometric analysis, epidemiological studies, automatic discovery of latent topics and causal relationships across multiple domains of the economy (UN Global Pulse,

2012, 2016; Jennifer McNabb & Puckett, 2009; Krishnamurthy & Desouza, 2014; Varian, 2014; Einav & Levin, 2014, 2013; Athey & Imbens, 2015; Kleinberg et al. , 2015; McBride & Nichols, 2015). Indeed, extending the scope of Aadhaar from just being an identification and authentication system for social welfare schemes to a system which generates large scale data and facilitates automated analysis and planning, can potentially lead to far reaching benefits.

At the same time, apart from the concerns of loss of privacy and civil liberties, the Aadhaar project has attracted considerable criticism for causing significant disruptions and exclusions in social welfare schemes (Johari, 2016; NDTV, 2016b,a; Dréze, 2016; Yadav, 2016a,b), both due to careless deployment and uncertainties in biometric matching.

We believe that all the above issues, both for and against, require careful analysis and rigorous evaluation; and that the technological, legal and policy frameworks need to be considerably strengthened through debates and informed choices to evolve an effective national identity scheme.

1.3 Privacy and security concerns

We examine the following main concerns pertaining to privacy and security in Aadhaar:

1. Identification of individuals without consent using the global Aadhaar number.
2. Identification and authentication without consent using demographic and biometric data.
3. Surveillance, tracking or profiling of people beyond legal sanctions using the centralised database, either through external hacks or through insider leaks and collusion.

Specifically, we ask the following questions which we believe are crucial for ensuring safety of Aadhaar:

1. Is it possible to ensure that user data and identification and authentication trails are completely protected from manual inspection by the UIDAI or the Government or any other entity or individual, thereby effectively preventing unauthorised surveillance?
2. Is it possible to ensure that all transactions, investigations and analytics can be carried out in a safe way only through audited, pre-approved and tamper proof computer programs? Additionally, can it be ensured that the programs are true to legal and policy frameworks, do precisely and only what they are supposed to do, and maintain tamper proof logs of all authorisation chains and results?

We believe that the above questions capture the essence of privacy protection in computerised databases. Privacy protection does not demand that data should not be collected, stored or used, but that there should be provable guarantees that the data cannot be used for any purpose other than those that have been approved.

1.4 Our Goal

Recent advances in Computer Science offer several novel and powerful solution ideas to address many of the privacy and security challenges posed by the project. Our goal is to carefully examine the security concerns, survey the technological tools that may aid us and provide a first order analysis of what might be feasible.

Our approach is as follows. We first capture the functionality desired by the Aadhaar project. Next, we analyse the security risks and vulnerabilities engendered by each entity and each communication link in the Aadhaar model. We examine the security measures proposed by UIDAI and discuss where these may be lacking. We elucidate recent tools from computer science, particularly from the fields of cryptography and

security, which may assist in providing safeguards: this puts some stated concerns to rest while simultaneously raising multiple unforeseen issues. We summarise our key findings and recommendations in Table 1 in the concluding section (Section 7).

Overall, we hope that our work provides a rigorous and scientific treatment of privacy concerns regarding Aadhaar, and enables well informed and well reasoned decisions regarding deployment.

The rest of the paper is organised as follows. In Section 2 we describe the Aadhaar functional architecture, and the various entities involved and their roles; in Section 3 we analyse the privacy and security requirements of the Aadhaar project. In Section 4 we discuss the subtle differences between identity verification and authentication and point out that failure to demarcate the two may lead to authentication without consent. In Section 5 we analyse the possibilities of privacy breaches through the Aadhaar number and suggest possible remedies. In Section 6 we analyse the threats for potential privacy breaches from the Aadhaar database and the field devices and explore possible approaches that may be adopted to mitigate the risks. Finally, in Section 7 we conclude the paper.

2 The Aadhaar model

In this section we describe the various entities involved in Aadhaar and their inter-dependencies, which will enable us to reason about its privacy and security requirements. The Aadhaar authentication and identity verification system comprises the following entities (UIDAI, 2016b):

1. The *Unique Identification Authority of India* (UIDAI) is responsible for providing the basic identification and authentication services. It provides a unique identifier (Aadhaar number) to each resident and maintains their biometric and demographic data in a *Central Identities Data Repository (CIDR)*. The UIDAI manages the CIDR and provides identification and authentication services with yes/no answers.

2. An *Authentication User Agency (AUA)* who provides services to users that are successfully authenticated. Thus, an AUA connects to the CIDR and uses Aadhaar authentication to validate a user and enable its services. Examples of AUAs and services are banks, various state and central government ministries providing services such as the Public Distribution System (PDS), the National Rural Employment Guarantee Scheme (NREGS), and even private agencies like mobile phone operators. The responsibility of logistics of service delivery rests with the AUAs. In this federated model an AUA may choose to use only Aadhaar identification, or also authentication in conjunction with their own legacy identification and authentication systems. An AUA is required to enter in to a formal contract with UIDAI to be able to use Aadhaar authentication services.

3. An *Authentication Service Agency (ASA)* is an entity that has a secure leased line connectivity with the CIDR. ASAs transmit authentication requests to CIDR on behalf of one or more AUAs. An ASA enters into a formal contract with UIDAI.

4. The *users*, namely, the residents of the country who enrol themselves with UIDAI and are issued unique identification numbers (Aadhaar numbers). A user has to present this number as the basic identification to an AUA for availing Aadhaar authentication services. The Aadhaar number for a user is common across all AUAs and service domains.

5. The *Point of Sale* (POS) device, also known as authentication device which collects personal identity data from Aadhaar holders, prepares the information for transmission, transmits the authentication

packets for authentication and receives the authentication results.

6. An *Enrolment Station*, which is a collection of field devices used by enrolment agencies appointed by UIDAI to enrol people in to the Aadhaar database and capture their demographic and biometric particulars.

The Aadhaar number is common across all AUAs and service domains. The framework (without the enrolment station) is captured in Figure 1.

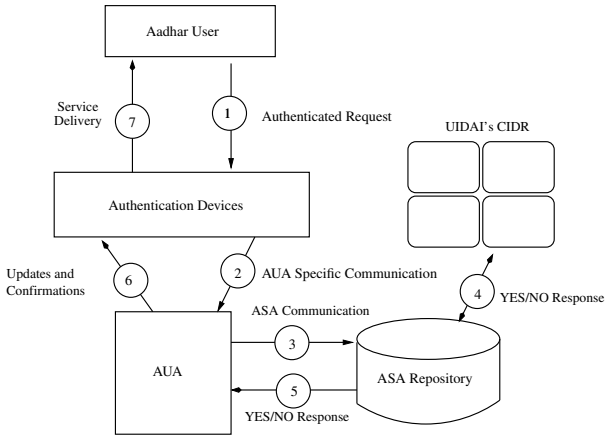

Figure 1: The Aadhaar authentication framework (Figure inspired from (UIDAI, 2016b))

3 Privacy and security in Aadhaar: definitions, assumptions and requirements

In this section we do a requirement analysis for privacy and security. To begin, we provide some definitions.

3.1 Identity verification vs authentication

Aadhaar is a national identity project, but we believe that the subtle difference between identity verification and authentication is itself not well understood, and this leads to confusions in policy making and deployment. Below, we attempt to first demarcate the two concepts.

According to standard notions of digital authentication, a security principal (a user or a computer), while requesting access to a service, must provide two independent pieces of information - *identity* and *authentication*. Whereas identity provides an answer to the question "*who are you?*", authentication is a challenge-response process that provides a "*proof of the claim of identity*", typically using an authentication credential. Common examples of identity are User ID (Login ID), cryptographic public keys, email ids, ATM or smart cards; some common authentication credentials are passwords (including OTPs), PINs and cryptographic private keys.

Identity may be considered public information but an authentication credential must necessarily be private - a secret that is known only to the user. Moreover, authentication must be a conscious process that requires active participation by a user, but not necessarily so for identity verification. As example use cases,

a bank may want an identity verification while opening an account at which stage no secret like a password is usually necessary, but a user needs to authenticate with a PIN for transactions like ATM withdrawals. No publicly known information should be used as an authentication credential.

3.2 Privacy protection: fundamental assumptions

To determine the extent to which security and privacy are achieved, we must first define the desired expectations in this context. Our analysis is based on the following assumptions, which we believe are fundamental:

1. Authentication without consent should never be possible under any circumstances. Identification without consent should also not be possible except in some special situations like disaster management, identification of accident victims, law enforcement and such others. It should be noted that providing one's identity for obtaining services in any local context is always with consent.

2. Unapproved profiling, tracking and surveillance of individuals should not be possible. There should be sufficiently strong measures to prevent such breaches in privacy, with user-verifiable proof of the same.

3. The technical implementation of privacy and security must be provably correct with respect to the legal framework. The legal framework, in turn, needs to be suitably enhanced with special provisions to protect the privacy of individuals and society in an advanced information technology setting.

3.3 Possible ways of breach of privacy

In what follows we briefly examine the various ways in which the privacy of an individual can be compromised in a setting such as in Aadhaar.

1. *Correlation of identities across domains:* It may become possible to track an individual's activities across multiple domains of service (AUAs) using their global Aadhaar ids which are valid across these domains. This would lead to identification without consent.

2. *Identification without consent using Aadhaar data:* There may be unauthorised use of biometrics to illegally identify people. Such violations may include identifying people by inappropriate matching of fingerprint or iris scans or facial photographs stored in the Aadhaar database, or using the demographic data to identify people without their consent and beyond legal provisions.

3. *Illegal tracking of individuals:* Individuals may be tracked or put under surveillance without proper authorisation or legal sanction using the authentication and identification records and trails in the Aadhaar database, or in one or more AUA's databases. Such records will typically also contain information on the precise location, time and context of the authentication or identification, and the services availed.

We wish to emphasize that *insider attacks* are the most dangerous threats in this context. For instance, the second and third attacks above are much more likely if the attacker can collude with an insider with access to various components of the Aadhaar system.

3.4 Requirement analysis for privacy protection

In view of the above, effective privacy protection not only requires protecting the Aadhaar system from external attacks but from internal attacks as well. This requires strong guarantees on securing the data, logs and the transaction trails in the Aadhaar and the AUA systems. Specifically, one requires that:

1. UIDAI cannot be trusted against possible system hacks, insider leaks, and tampering of authentication records and audit trails. Indeed, the identity verification and authentication providing applications running on UIDAI computer systems should be trustworthy even when the UIDAI systems and the network cannot be trusted.

2. Manual inspection of user data, authentication records and audit trails should not be possible. In special cases of properly authorised investigations, such inspections may only be possible through pre-approved, audited and provably tamper proof computer programs, and an accurate tamper proof record of the entire investigation and digitally signed authorisation chain must be maintained at all times.

3. The enrolment agencies and the enrolment devices cannot be trusted from data privacy and security points of view; neither can the Point of Sale (POS) devices and various authentication user agencies (AUAs), whether government or private, be trusted for data protection.

4. Authentication User Agencies (AUAs) cannot be trusted with biometric and demographic data; neither can they be trusted with sensitive user data of private nature (for example, medical and immunisation records etc.). All provisions of data privacy and security that apply to UIDAI, must also apply to the AUAs. Strong legal and policy frameworks are required to ensure this.

5. It should not be possible to correlate identities across application domains, except on suitably anonymised data through pre-approved, audited and provably tamper proof computer programs for carrying out data analysis.

In what follows we discuss the various threats and vulnerabilities that result from the Aadhaar project in more detail and analyse the measures adopted by the UIDAI against these. We also suggest a few possibilities of enhancing the privacy and security protections.

4 Authentication without consent

As we have already discussed, authentication without consent should not be possible under any circumstances. Additionally, it should be possible to revoke an authentication credential in case it is compromised, with the identity of the individual remaining intact.

UIDAI defines Aadhaar authentication as follows (UIDAI, 2016a): *"Aadhaar authentication is the process wherein Aadhaar number, along with other attributes (demographic/biometrics/OTP) is submitted to UIDAI's Central Identities Data Repository (CIDR) for verification; the CIDR verifies whether the data submitted matches the data available in CIDR and responds with a Yes/No. No personal identity information is returned as part of the response"*. (UIDAI, 2016a) goes on to define five types of Aadhaar based authentication:

1. *Type 1 Authentication:* Through this offering, service delivery agencies can use Aadhaar Authentication system for matching Aadhaar number and the demographic attributes (name, address, date of birth, etc) of a resident.

2. *Type 2 Authentication:* This offering allows service delivery agencies to authenticate residents through One-Time-Password (OTP) delivered to resident's mobile number and/or email address present in CIDR.

3. *Type 3 Authentication:* Through this offering, service delivery agencies can authenticate residents using one of the biometric modalities, either iris or fingerprint.

4. *Type 4 Authentication:* This is a 2-factor authentication offering with OTP as one factor and biometrics (either iris or fingerprint) as the second factor for authenticating residents.

5. *Type 5 Authentication:* This offering allows service delivery agencies to use OTP, fingerprint and iris together for authenticating residents.

4.1 Analysis of UIDAI measures

Thus, we see that authentication is implemented in Aadhaar via the mechanisms of passwords and biometric information. However, in the usage of biometrics, we believe there is an implicit confusion between the concepts of identity verification and authentication. In the above usage, biometric information is used for authentication relying on the unstated assumption that this information is private. However, we argue that biometric data is public: for instance, people's fingerprints can be lifted without their consent from a variety of objects that they may touch and their iris data may be picked up by a high resolution, directional camera from a distance. Even DNA information can be obtained from the objects that users may touch (Houck & Houck, 2008). Hence fraudulent presentation of biometric data for authentication, without conscious participation by a user, is a definite possibility (Akhtar, 2012).

Another difficulty with using biometrics as authentication credentials is that revoking biometrics like fingerprints or iris for a compromised user is problematic[1].

4.2 Possible remedies

The analysis in the prior section leads us to conclude that the usage of only biometrics in the context of Aadhaar authentication (type 3 authentication above) has significant problems. Type 1 authentication is susceptible to the same problem, since it also uses public information for authentication. It will be necessary to use other factors, like trustworthy manual oversight, in conjunction with these modalities for authentication. The other types use at least one private modality, and are hence safe.

We note that biometrics can certainly be very useful for identity verification. A careful case analysis must be performed to delineate whether identity verification or authentication is required in any given context, and UIDAI should appropriately change its authentication architecture to account for the above. Also, the legal and policy frameworks must make a clear distinction between authentication and identity verification.

5 The Aadhaar number and the possibility of identification without consent

The Aadhaar number is at the heart of the Aadhaar scheme and is one of the biggest causes of concern. Recall that the Aadhaar number is a single unique identifier that must function across multiple domains. Given that the Aadhaar number must necessarily be disclosed for obtaining services, it becomes publicly

[1]We note that there is a notion of cancellable biometrics, but this is still in the research domain (Patel et al. , 2015; Tulyakov et al. , 2005) and may not yet integrate well with commercial matching software.

available, not only electronically, but also often in human readable forms as well, thereby increasing the risk that service providers and other interested parties may be able to profile users across multiple service domains. Once the Aadhaar number of an individual is (inevitably) known, that individual may be *identified without consent* across domains, leading to multiple breaches in privacy as discussed in (Makkar, 2016; Centre for Internet & Society, 2016; The London School of Economics and Political Science, 2005).

Another worrisome issue is that of *identity theft*, whose potential for damage now increases manifold. As an illustrative example, let us consider the US Social Security Number (SSN) (Wikipedia, 2016s). The primary difference between Aadhaar and SSN is that the SSN does not have any biometric identifier attached and it does not support authentication. The SSN associated with a person provides a single interface to the person's dealings with a vast number of public and private bodies, very similar to how the usage of the Aadhaar number is being envisaged. While this facilitates use of administrative data for useful data analytics (Jennifer McNabb & Puckett, 2009), the ease of obtaining the SSN from across public and private databases also results in extremely high number of identity theft cases in the US (The London School of Economics and Political Science, 2005, page 100).

5.1 Analysis of UIDAI measures

The UIDAI does acknowledge the possibility of breach of privacy that can arise due to the use of a single identifier across multiple domains and recommends that the AUAs should use only domain specific identifiers in their dealings with people (UIDAI, 2011, page 7). Examples of domain specific identifiers are bank account numbers, passport numbers, driving license numbers, ration card numbers etc. The UIDAI mandates that the AUAs should maintain a mapping between their domain specific identifiers and the global Aadhaar numbers at their back-end. The UIDAI does not maintain any such mapping and assumes that there cannot be any breach of privacy from the UIDAI because the mappings are unidirectional.

This however does not fully mitigate the risks and the possibility of leakage of the Aadhaar number from an AUA, either from the database, or during "Know Your Customer" (KYC) processes, or even during availing services, cannot be ruled out. In particular, there appear to be no safeguards or even guidelines, either technical or legal, on how the Aadhaar number should be maintained and used by various AUAs in a cryptographically secure way, and how to prevent the Aadhaar number of an individual from becoming public. In fact, in many of the schemes that require Aadhaar authentication, it is necessary to provide the Aadhaar number as a public identifier. With such weak provisions, identification without consent and correlation of identities across application domains without approval remain as real possibilities. Additionally, since the Aadhaar number is supposed to be valid for life (UIDAI, 2011), it cannot easily be revoked in case of an identity theft or if the Aadhaar number is compromised in any other way.

5.2 Possible remedy

Thus, linking individuals across domains with a global identifier for legitimate data analysis, and the possible loss of privacy because of the correlation of identity across domains such a global identifier facilitates, are conflicting requirements. An alternative and more principled strategy to resolve the conflict would be for the UIDAI to issue different local identifiers (different Aadhaar numbers) for different domains, but to cryptographically embed in to all local identifiers a unique "master identifier". Several alternatives are possible. One may design the identifiers so that no linking across domains is possible at all, and it is impossible to isolate the global signature from any of the local identifiers. The linking then becomes unidirectional, but in the reverse direction to what UIDAI has currently suggested. Alternatively one may allow limited linking across domains, either bi-directional or even uni-directional. The LSE identity report actually suggests

such a scheme (The London School of Economics and Political Science, 2005). Correlation across multiple domains using the master identifier, through cryptographically secure and pre-approved data analytics software, will always be possible in such a scheme. Sufficiently strong cryptographic measures should be used to embed the master identifier in to the local ones to prevent against possible external correlation attacks. Also, a major shift in the policy framework is necessary to reverse the direction of linking.

6 Protection of user data

In Section 2 we discussed that a major threat to privacy of users arises from the possibility of insider attacks. In this section we discuss the possibilities of securing Aadhaar from such threats. While specifying the security requirement of a scheme such as Aadhaar - involving clients, service providers and an authentication agency - one can assume that standard measures against external attacks like network firewalls, intrusion detection, hardware security modules (HSM), operating system hardening etc., will be incorporated by default.

6.1 Threat levels

In what follows, we outline the various levels of threat that are possible, and measures that can be taken in each case.

6.1.1 Untrusted network, trusted application and authentication servers, trusted clients

Among others, this scenario is common in internet banking, where the application and authentication servers are usually the same; in campus networks where snooping and attacks are fairly common; and in various internet and mobile application based services that use Google or Facebook for authentication. The basic security requirements in such situations are

- The authentication servers and the application servers must authenticate themselves to each other, and to the clients, to prevent against possible man-in-the-middle attacks (Wikipedia, 2016i).
- User credentials and other critical data must never travel over the network in unencrypted form.

Some commonly used solutions are provided by the encrypted version of the hypertext transfer protocol (HTTPS) (Wikipedia, 2016f) for accessing secure and signed web-pages, secure shell (SSH) (Wikipedia, 2016p) for establishing secure terminal sessions and virtual private networks (VPN) (Wikipedia, 2016v) for securely accessing an enterprise's private network from a remote location. All these protocols use encrypted tunnels through untrusted networks using techniques such as public key cryptography (PKI) (Wikipedia, 2016m), which enable the clients and the servers to verify the authenticity of other public servers, and possibly each other, before establishing secure encrypted communication channels. Kerberos (Wikipedia, 2016g) (also used in Microsoft Active Directory) based authentication and the Radius protocol (Wikipedia, 2016n) for WiFi are often used in campuses; and OAuth (Wikipedia, 2016l) is a common authentication and authorisation framework for public applications that use Google or Facebook for authentication. The client (access devices) and the servers themselves are assumed to be trusted and safe.

6.1.2 Untrusted network, untrusted authentication and application servers, trusted clients

This is a more challenging security situation where, in addition to the above, one also has to worry about data leaks from the servers, either due to hacking or even due to insider leaks. Some common counter measures are

- The authentication servers must never store any user credentials and may only store a Hash (Wikipedia, 2016c), a value computed from user credential(s) using a non-invertible function, and use it for matching. Then, user credentials can never leak.
- All critical data, records and logs must be stored only in encrypted forms on the servers. The decryption keys should not be easily accessible.
- There must be provisions for tamper detection for both data and programs.

Some of the common solutions are provided by cryptographic hashing techniques like *Secure Hash Algorithms (SHA-n)* (Wikipedia, 2016o,c), encryption techniques based on *Advanced Encryption Standard (AES)* (Wikipedia, 2016b) for protecting data, credentials and logs, *keyed-hash message authentication code (HMAC)* (Wikipedia, 2016e) based tamper detection, and hardware security modules (HSM) (Wikipedia, 2016d) for restricting insider access. These methods typically use symmetric key cryptography where the same key is used for both encryption and decryption. Challenge-response based authentication protocols like Kerberos (Wikipedia, 2016g) can ensure that the user credentials never travel over the network, even in encrypted format.

In even stricter situations, one may require in addition that

- The authentication servers must never store any information about user credentials, not even a hash.
- No process at the authentication servers should be able to glean any information whatsoever about user credentials from the information exchange during an authentication process.
- There are stronger guarantees for tamper detection. In particular,
    1. The authentication and other servers must be able to prove to any designated auditor that they have not been tampered with and are running only pre-approved and inspected computer programs.
    2. The servers must also be able to prove that none of their data, including records and log files, have been manually inspected or modified.

Advanced cryptographic measures may be used for such authentication (Wikipedia, 2016w; Sakashita et al. , 2009), although their usage is not common. Neither is it common to demand such strong tamper proof guarantees in most authentication and application systems. In Section 6.3 we discuss some possibilities for such strong tamper protection guarantees.

6.1.3 Untrusted clients

In almost all internet applications, including banking, it is tacitly assumed that the client access devices - mobiles and handhelds, laptops and desktop computers - are trusted, and the responsibility of data protection in these devices are passed on to the users. However, in special situations where the access devices are not owned by the users but are provided by service providers, the users may have a right to be assured that data and credentials cannot be compromised from the access devices. Examples of such access devices are ATM machines, Aadhaar enrolment stations and other POS terminals. In all such cases one may require that:

- A client terminal or a POS device must be able to prove at all times to the server, and also to any approved third party auditor, that it has not been tampered with and does only what it is supposed to do.
- It should also be able to provide such a proof to a discerning user.

Later, in Section 6.3.2 we discuss some possible solutions for securing such access devices.

6.2 Analysis of UIDAI measures

The security and privacy infrastructure of UIDAI has the following main features (UIDAI, 2014):

- 2048 bit PKI (Wikipedia, 2016m) encryption of biometric data in transit. End-to-end encryption from enrolment/POS to CIDR.
- Trusted network carriers (ASAs) between CIDR and AUAs. Effective precaution against denial of service (DOS) attacks.
- HMAC (Wikipedia, 2016e) based tamper detection of PID (Personal Identity Data) blocks, which encapsulate biometric and other data at the field devices.
- Registration and authentication of AUAs.
- Within CIDR only a SHA-n Hash (Wikipedia, 2016o) of Aadhaar number is stored.
- Audit trails are stored SHA-n encrypted (Wikipedia, 2016o), possibly also with HMAC (Wikipedia, 2016e) based tamper detection.
- Only hashes of passwords and PINs are stored. Biometric data are stored in original form though.
- Authentication requests have unique session keys and HMAC (Wikipedia, 2016e). Protection against replay attacks.
- Resident data stored using 100 way sharding (vertical partitioning) (Wikipedia, 2016q). First two digits of Aadhaar number are used as shard keys.
- All enrolment and update requests link to partitioned databases using RefIDs (coded indices).
- All system accesses, including administration, through a hardware security module (HSM) (Wikipedia, 2016d) which maintains an audit trail.
- All analytics carried out only on anonymised data.

While these measures appear to be quite reasonable against external attacks, they may not be enough to forestall insider attacks. Though the safeguards adequately address the threat scenario described in Section 6.1.1, they are not adequate for the threat levels described in Sections 6.1.2 and 6.1.3. For something as important as the national identity project, one will have to assume that the biggest security and privacy threats come from insider leaks. These include possible unauthorised and surreptitious examination of data, transaction records, logs and audit trails by personnel with access, leading to profiling and surveillance of targeted groups and individuals, perhaps at the behest of interested and influential parties in the state machinery itself. Hence, one would ideally like to have provisions to guard against the threat levels described in Section 6.1.2 and Section 6.1.3.

The following appear to be the main weaknesses:

1. Most of the security measures are based on cryptographic encryption techniques that require cryptographic keys to decode. Protection of these keys is of great importance, and it is necessary to have suitable measures to do so. Currently, we do not find mention of any such measures, and we believe that assuming trust in this context is a significant vulnerability. We do not believe that hardware security modules (HSM) (Wikipedia, 2016d), which are also under the administrative control of the same organisation, offer adequate protection against insider attacks for something as crucial as the national identity verification and authentication system.

2. There appears to be no well defined and cryptographically sound approval procedure for data inspection, whether for investigation or for analytics. This makes the system extremely open to abuse.

3. There appears to be no well defined procedure for audit and approval of various UIDAI programs and software. In particular, one would like to be able to establish that the programs have not been tampered with, and are doing precisely what they are supposed to do.

4. There appears to be no proper tamper detection and runtime audit of the field devices, including enrolment stations, to ensure that they are functioning true to specifications, and that there is no possibility of data leakage from the field devices.

5. Finally, we note that user biometric data are stored in the central repository, perhaps encrypted, but this still violates an important safeguard that we mentioned in Section 6.1.2 - that user credentials should never be stored on the server. Unless there are some specific reasons to store the original biometric data it may be safer to store only non-invertible intermediate representations which are sufficient for matching (Tulyakov et al. , 2005; Dodis et al. , 2004).

6.3 Possible measures against insider attacks

Our starting point is that the environment in which the CIDR programs (code) are executed cannot be assumed as trusted. One must address the possibility that the attacker has full access to the computer programs that may be running on the UIDAI database. This may include both the source code and the runtime environment. How can one hope to secure such a system against insider attacks? We believe that two independent lines of defence are required:

1. There has to be an independent third party that can play the roles of an auditor and a keeper of cryptographic keys.

2. Several modern tools and techniques from computer science offer (partial) solutions to these problems. These need to be studied, evaluated and appropriately deployed.

In what follows we briefly describe each of the these.

6.3.1 Need for an independent third party

Note that although critical data and transaction logs are maintained encrypted within the UIDAI, the decryption keys are also stored in the UIDAI systems. Since the decryption must happen routinely, the computer programs running in the UIDAI systems must be able to access these keys. There is no reason to believe that these keys cannot be retrieved with the collusion of multiple parties within the UIDAI in which case the data may be illegally accessed. The data may even be tampered and the HMAC (Wikipedia, 2016e) signatures recomputed without leaving a trail. Also, there is no way to guarantee that no unauthorised or

tampered computer programs are running on the UIDAI systems. HSMs (Wikipedia, 2016d) under the same administrative control are not sufficient measures to engender confidence.

The problem appears to be unsolvable without the introduction of a completely independent third party, under a different administrative control, that can play the roles of a key-keeper and an auditor.

Distributed key management. At least a part of every crucial decryption key must remain with the third party and a distributed key management protocol (Wikipedia, 2016h) must be put in place. The third party must programmatically share the portion(s) of the key(s) it holds with a corresponding computer program in the CIDR at run-time, through a cryptographically secure channel, only after authenticating the genuineness of the program using a cryptographic certificate and verifying that the program has not been tampered with.

Audit and approval of UIDAI programs. To enable the above, it will be necessary for the auditor to examine, approve and cryptographically sign every program that may run in the CIDR. Thereafter, these programs should periodically during run-time and on demand, cryptographically prove to the auditor's programs that they are genuine and have not been tampered with.

Audit of data inspection. All data inspection, including those through special purpose programs for data analytics, should be digitally approved by the auditor.

There has to be proper legal provisions for setting up such a third party audit and key-management system.

6.3.2 Tools and techniques from computer science

Even with the above measures in place, the complete decryption keys will have to reside in the memory of the UIDAI computer systems at some point during the run-time. A well trained system administrator, with access to the hardware and the operating system, will still be able to access the decryption keys from the system's memory. There are a variety of tools in computer science that may provide a defence against such run-time attacks. We describe some of them below.

Storing Hash of biometric data. Since the Aadhaar database stores sensitive biometric data of individuals, a useful strategy to protect this data is to store only a non-invertible hash of biometric data, which converts a string representing biometric data to a nearly uniform random string which does not leak any information about the individual. Some techniques to achieve these are *fuzzy extractor* (Dodis et al. , 2004) and symmetric hashing (Tulyakov et al. , 2005).

Tamper-proof code. A significant cause of concern is that a malicious insider may be able to modify the code so that it behaves arbitrarily. Such attacks are dangerous not just in terms of denial of service, but also because arbitrary behaviour may lead to leakage of secrets embedded in code.

To address these attacks, one may use available techniques to transform code to *tamper-resistant* code (Wang et al. , 2000; Michiels & Gorissen, 2007). Additionally, there are methods which prove the security and integrity of code without code transformations. Knowing that the behaviour of the server code does not change, one can resort to standard *static-code* analysis (Wikipedia, 2016t) and *model checking* techniques (Wikipedia, 2016j) in order to verify whether the server code works according to intended specifications. The behavioural specifications can be coded in known and industry-adopted formal executable languages

(such as TLA+, Simulink/Stateflow, Message Sequence Chats, *etc*.). However, performing end-to-end static-code analysis or model checking on server code only addresses the question whether or not server code is acting maliciously. In the scenario when server code is established to be kosher, one has to still address the question of security when an adversary (read as environment or external user) tries to modify the code by cunningly exploiting the input interface of the server. A promising solution proposed in the security literature, which appears to be address the afore-mentioned problem is *control-flow integrity* (Martin Abadi, 2005). In a nutshell, this involves a runtime check on the code by installing an additional *monitor* which tracks the execution behaviour of the code and is aware of admissible behaviours of the program apriori. Third party audit will be required to set up the processes to ensure that the code is tamper free.

Tamper-proof hardware. In addition to software solutions, tamper resistant hardware may also be leveraged for protection of cryptographic keys or data. For instance, severs in the presence of non-volatile memories can be subjected to side-channel attacks (Wikipedia, 2016r). Sensitive data that resides in memory across reboots can leak out to attackers. One way to secure a server's memory is to encrypt all memory transfers that take place between the processor cache onwards and the main memory. Encryption mechanisms as discussed in prior sections could be utilised for memory-transfer encryptions. A separate encryption controller unit would be mandated for this purpose.

Intel's Software Guard Extension (SGX) (Costan & Devadas, 2016) and its forerunners TPM and TXT (Wikipedia, 2016u) are yet another handy off-the-shelf solutions where protected areas (enclaves) of hardware are provided where execution of applications can take place without compromising BIOS, drivers, memory buses, and application's security. Furthermore, SGX also provide solutions for remote attestation challenges to ensure hardware integrity. Trusted hardware may be leveraged to provide sought integrality and confidentiality. Here again, setting up and the safe-keep of the trusted hardware has to be entrusted to a third party organisation different from the UIDAI.

Secure multi-party computation. Another method to secure keys or other private inputs is offered by the field of secure multi-party computation. Secure multi-party computation (Wikipedia, 2016k) is a field of cryptography that allows several mutually distrustful parties, each wishing to maintain privacy of their input data, to perform some computation on their joint data. This is a rich field with several efficient mechanisms in place to perform a large class of interesting computations privately. We believe that tools and techniques from this field may be relevant to the Aadhaar project: for instance, one may use a secret-sharing scheme to split the database across two servers belonging to different entities, ensuring that the two servers have disjoint sets of system administrators and diverse operating systems and hardware. This ensures that even if one server is hacked into, the data remains protected. Secure multiparty computation can be used to answer queries on the data distributed across servers.

Homomorphic and functional encryption. Another security threat is the possibility of server breaches, whether the attack is launched from inside or outside the organisation. To prevent a server breach from leaking valuable user data, critical data needs to be stored on the server in encrypted form. However, encrypting data using standard methods leads to loss of functionality, such as the ability to perform data analytics. Recently, advanced forms of encryption have been designed by the cryptographic community that allow an untrusted server to compute on data "blind-folded". Two striking examples of such encryption mechanisms are the notions of homomorphic encryption (Gentry, 2009) and functional encryption (Sahai & Waters, 2005). At a high level, these systems allow sensitive data to be encrypted in a way that allows

sophisticated computation on the data *in its encrypted form*. Thus, the functionality offered by data analytics can be enjoyed while ensuring privacy.

Such mechanisms may be very pertinent to ensuring privacy of data in the UIDAI database. However, while these systems are substantial achievements in cryptographic design, they remain far too slow for practical use. Nevertheless, for restricted classes of computations, such algorithms may be deployed. Third party intervention will be required to set up the computation in the encrypted domain.

Symmetric Searchable Encryption and Extensions. Another method to perform useful computations on encrypted data is offered by the field of symmetric searchable encryption, which enables searching on encrypted data (Bellare et al. , 2007; Curtmola et al. , 2011). Unlike notions such as functional encryption and homomorphic encryption described above, algorithms developed in the context of searchable encryption are highly efficient and scale well for massive sized data, such as the UIDAI data. For many investigative applications, tools and techniques developed in the context of searchable encryption appear to be very relevant.

Whiteboxing and code obfuscation. Another useful class of defenses against insider attacks comes from techniques developed in the area of whitebox cryptography. Typically, one assumes that attacks are *black-box*, i.e., an attacker has access to the input and the output of a program, but not to the internal workings of the program. However, an insider may have full access to the source code and binary file running on the system, and also the corresponding memory pages during execution. Additionally, the attacker can also possibly make use for debuggers and emulators, intercept system calls and tamper with the binary and its execution. Such attacks are called *whitebox* attacks, and whitebox cryptography (Preneel & Wyseur, 2008) aims to implement cryptographic procedures in software that transform and obfuscate code and data in such a way so that the cryptographic assets remain secure even when subject to whitebox attacks.

Although whitebox cryptography and obfuscation have been plagued with numerous attacks and there are impossibility results in theory for the general problem (Barak et al. , 2001; Billet et al. , 2004; Wyseur et al. , 2007), successful whiteboxing in specific situations may well be possible (Delerablée et al. , 2014; Bogdanov & Isobe, 2015). Many software packages that provide whitebox protection in restricted scenarios are available, and despite the lack of rigorous cryptographic guarantees, seem to work well in practice. Such packages may be deployed to enhance security against insider attacks. Note that the whitebox protection of security keys and the decryption code will have to be put in place by an independent third party.

6.4 Securing field devices

Finally, client access devices (or POS devices) can broadly be understood to have the same critical components that CIDR servers have: hardware (the device itself), and the application(s) running on the device. Solutions to secure client devices are no different than the solutions for servers that we discussed above.

7 Conclusions

We have analysed the Aadhaar project from the points of view of privacy and security and have pointed out some technical weaknesses and possible remedies. Specifically, we have found that

1. The Aadhaar number, which is a single global identifier that is supposed to work across application domains, makes individuals vulnerable to privacy breaches. A design alteration can however make it safe.

Issue	Shortcoming in UIDAI measures	Key recommendations
Authentication without consent	• Biometric and demographic data are public; hence, can be used without consent	• Demarcate identity verification and authentication. • Strengthen legal and policy frameworks See Sections 3 and 4 for details.
Identification without consent using Aadhaar number	• Unidirectional linking from AUA-specific local ids to Aadhaar id. • No guidelines on safe maintenance of Aadhaar numbers by AUAs. • Vulnerable to correlation of identity across domains.	• Unidirectional linking from Aadhaar id to AUA-specific ids • Cryptographically embed Aadhaar id into AUA-specific ids making correlation impossible See Section 5 for details.
Unlawful access of CIDR data leading to profiling, tracking and surveillance	• Inadequate protection against *insider* attacks on CIDR data • CIDR data encrypted but the decryption keys reside in CIDR • UIDAI human managers can have access to decryption keys	• Separate administrative control for online audit and key management • Legal framework for the above • Only hashes of biometric data must be stored on servers. • Manual inspection of CIDR data must not be possible • Only pre-approved and audited computer programs with tamper-proof guarantees should access CIDR data • All investigations and analyses only with prior audit and approval through pre-approved computer programs • Tamper proof guarantees for field devices • Adopt modern tools from computer science to implement the above protections See Section 6 for details.

Table 1: Summary of our analysis and recommendations

2. The slightly different concepts of authentication and identity verification need to be well demarcated, and careful use case analysis is required to determine precisely what is required for each application. The legal framework must also make note of these.

3. In an Aadhaar like setup, the biggest threat to privacy comes from potential insider leaks. The Aadhaar technology architecture does not seem to have been explicitly designed to have strong protections against such insider leaks. We believe that effective protection against insider leaks necessarily requires a third party auditor under independent administrative control. With such a provision in place there are several tools from computer science that can provide reasonable guarantees for security and privacy protection.

We summarise our analysis and key findings in Table 1.

Thus, though there are serious privacy concerns at present, we believe that Aadhaar can be made safe from a technology perspective with due-diligence. The legal framework however needs to be more specific and requires significant strengthening. Perhaps the single most important specific question that begs answering is *who should have the right to verify the identity of an individual, and under what circumstances?* Above

all, we believe that the Aadhaar project requires informed and comprehensive policy debates, covering all angles, to realise its full effectiveness without causing the kind of disruptions that have been reported.

The effectiveness of biometric identification and to what extent are the biometric features required are remaining important questions that require further study.

Acknowledgement

We thank Reetika Khera for the many discussions, and Ambuj Sagar and Narayanan Kurur for suggestions on improving our manuscript. The first author thanks Manoj Prabhakaran for many helpful comments and Mihir Bellare for suggesting the use of fuzzy extractors.

References

Akhtar, Zahid. 2012 (March). Security of Multimodal Biometric Systems against Spoof Attacks. Ph.D. thesis, Dept. of Electrical and Electronic Engineering, University of Cagliari. http://pralab.diee.unica.it/sites/default/files/Akhtar_PhD2012.pdf.

Arun, Chinmayi. 2016. Privacy is a fundamental right. http://www.thehindu.com/opinion/lead/lead-article-on-aadhaar-bill-by-chinmayi-arun-privacy-is-a-fundamental-right/article8366413.ece. [Online; posted 18-March-2016].

Athey, Susan, & Imbens, Guido W. 2015 (April). Machine Learning for Estimating Heretogeneous Casual Effects. Tech. rept. Stanford University. https://www.gsb.stanford.edu/faculty-research/working-papers/machine-learning-estimating-heretogeneous-casual-effects.

Barak, Boaz, Goldreich, Oded, Impagliazzo, Rusell, Rudich, Steven, Sahai, Amit, Vadhan, Salil, & Yang, Ke. 2001. On the (im) possibility of obfuscating programs. Pages 1–18 of: Annual International Cryptology Conference. Springer.

Bellare, Mihir, Boldyreva, Alexandra, & ONeill, Adam. 2007. Deterministic and efficiently searchable encryption. Pages 535–552 of: Annual International Cryptology Conference. Springer.

Bhatia, Gautam. 2015. Sorry, Mr. Attorney-General, We Do Actually Have a Constitutional Right to Privacy. http://thewire.in/2015/07/28/sorry-mr-attorney-general-we-do-actually-have-a-constitutional-right-to-privacy-7398/. [Online; posted 28-July-2015].

Billet, Olivier, Gilbert, Henri, & Ech-Chatbi, Charaf. 2004. Cryptanalysis of a white box AES implementation. Pages 227–240 of: International Workshop on Selected Areas in Cryptography. Springer.

Bogdanov, Andrey, & Isobe, Takanori. 2015. White-Box Cryptography Revisited: Space-Hard Ciphers. Pages 1058–1069 of: Proceedings of the 22Nd ACM SIGSAC Conference on Computer and Communications Security. CCS '15. New York, NY, USA: ACM.

Centre for Internet & Society, The. 2016. List of Recommendations on the Aadhaar Bill, 2016 - Letter Submitted to the Members of Parliament. http://cis-india.org/internet-governance/blog/list-of-recommendations-on-the-aadhaar-bill-2016. [Online; posted 16-March-2016].

Costan, Victor, & Devadas, Srinivas. 2016. Intel SGX Explained. IACR Cryptology ePrint Archive, **2016**, 86.

Curtmola, Reza, Garay, Juan, Kamara, Seny, & Ostrovsky, Rafail. 2011. Searchable symmetric encryption: improved definitions and efficient constructions. Journal of Computer Security, **19**(5), 895–934.

Delerablée, Cécile, Lepoint, Tancrède, Paillier, Pascal, & Rivain, Matthieu. 2014. White-Box Security Notions for Symmetric Encryption Schemes.

Diffie, Whitfield, & Hellman, Martin E. 1979. Privacy and Authentication: An Introduction to Cryptography. Proceedings of the IEEE, **67**(March), 397–427. http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=1455525.

Dodis, Yevgeniy, Reyzin, Leonid, & Smith, Adam. 2004. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. Pages 523–540 of: International Conference on the Theory and Applications of Cryptographic Techniques. Springer.

Dréze, Jean. 2016. The Aadhaar coup. http://www.thehindu.com/opinion/lead/jean-dreze-on-aadhaar-mass-surveillance-data-collection/article8352912.ece. [Online; posted 15-March-2016].

Duggal, Pavan. 2011. Does the UID project infringe on privacy? http://www.business-standard.com/article/opinion/does-the-uid-project-infringe-on-privacy-111080300006_1.html. [Online; posted 3-August-2011].

Einav, Liran, & Levin, Jonathan. 2014. Economics in the age of big data. Science, **346**(6210). http://science.sciencemag.org/content/346/6210/1243089.

Einav, Liran, & Levin, Jonathan D. 2013 (May). The Data Revolution and Economic Analysis. Working Paper 19035. National Bureau of Economic Research. http://www.nber.org/papers/w19035.

Express News Service. 2016. Aadhar Bill passed in Lok Sabha, Opposition fears surveillance. http://indianexpress.com/article/india/india-news-india/aadhar-card-uid-bill-lok-sabha-arun-jaitley/. [Online; posted 12-March-2016].

Gentry, Craig. 2009. Fully homomorphic encryption using ideal lattices. Pages 169–178 of: STOC.

Houck, Max, & Houck, Lucy. 2008 (August). What is touch DNA? http://www.scientificamerican.com/article/experts-touch-dna-jonbenet-ramsey/. [Online; posted 28-July-2015].

Jayaram, Malavika. 2015. Aadhaar debate: Privacy is not an elitist concern it's the only way to secure equality. http://scroll.in/article/748043/aadhaar-debate-privacy-is-not-an-elitist-concern-its-the-only-way-to-secure-equality. [Online; posted 15-August-2015].

Jennifer McNabb, David Timmons, Jae Song, & Puckett, Carolyn. 2009. Uses of Administrative Data at the Social Security Administration. Social Security Bulletin, **16**. https://www.ssa.gov/policy/docs/ssb/v69n1/v69n1p75.html.

Johari, Aarefa. 2016. In drought-hit Saurashtra, poor internet network can often mean no food rations. http://scroll.in/article/810683/in-drought-hit-saurashtra-no-internet-can-often-mean-no-food-rations. [Online; posted 26-June-2016].

Khera, Reetika. 2011. The UID Project and Welfare Schemes. Economic and Political Weekly, **Vol. 46**(Issue No. 09).

Khera, Reetika. 2013. Lessons from the East Godavari pilot. http://www.thehindu.com/opinion/lead/lessons-from-the-east-godavari-pilot/article4603273.ece. [Online; posted 11-April-2013].

Khera, Reetika. 2015. Five Myths about Aadhar. http://www.outlookindia.com/website/story/five-myths-about-aadhar/295364. [Online; posted 18-September-2015].

Kleinberg, Jon, Ludwig, Jens, Mullainathan, Sendhil, & Obermeyer, Ziad. 2015. Prediction Policy Problems. American Economic Review, **105**(5), 491–95. http://www.aeaweb.org/articles?id=10.1257/aer.p20151023.

Krishnamurthy, Rashmi, & Desouza, Kevin C. 2014. Big Data Analytics: The Case of Social Security Administration. Information Policy, **19**(May), 165–178. http://ssrn.com/abstract=2757871.

Kumar, Ashwani. 2015. Privacy, a non-negotiable right. http://www.thehindu.com/opinion/lead/privacy-a-nonnegotiable-right/article7519148.ece. [Online; posted 10-August-2015].

Lok Sabha Secretariat: New Delhi. 2012. Report by the Parliamentary Standing Committee on Finance (2011-2012) chaired by Shri Yashwant Sinha. http://planningcommission.nic.in/reports/genrep/rep_privacy.pdf.

Makkar, Sahil. 2016. Aadhaar is actually surveillance tech: Sunil Abraham. http://www.business-standard.com/article/opinion/aadhaar-is-actually-surveillance-tech-sunil-abraham-116031200790_1.html. [Online; posted 12-March-2016].

Martin Abadi, Mihai Budiu, Úlfar Erlingsson Jay Ligatti. 2005 (nov). Control-Flow Integrity. Pages 340–353 of: ACM Conference on Computer and Communication Security (CCS).

Masiero, Silvia. 2015. PDS computerisation: What other states can learn from Kerala. http://www.ideasforindia.in/article.aspx?article_id=1474. [Online; posted 6-July-2015].

McBride, Linden, & Nichols, Austin. 2015 (January). Improved poverty targeting through machine learning: An application to the USAID Poverty Assessment Tools. Tech. rept. Economics That Really Matters, Charles H. Dyson School of Applied Economics and Management at Cornell University. http://www.econthatmatters.com/wp-content/uploads/2015/01/improvedtargeting_21jan2015.pdf.

Mehta, Pratap Bhanu. 2016. Privacy after Aadhaar. http://indianexpress.com/article/opinion/columns/privacy-after-aadhaar-money-bill-rajya-sabha-upa/. [Online; posted 26-March-2016].

Michiels, W., & Gorissen, P. 2007. Mechanism for Software Tamper Resistance: An Application of White-box Cryptography. In: Proceedings of the 2007 ACM Workshop on Digital Rights Management. DRM '07.

NDTV. 2016a. Truth vs Hype: Aadhaar's One Billion Challenge. http://www.ndtv.com/video/news/truth-vs-hype/truth-vs-hype-aadhaar-s-one-billion-challenge-411279. [Online; posted 9-April-2016].

NDTV. 2016b. What should they do who dont get ration. http://khabar.ndtv.com/video/show/ndtv-special-ndtv-india/what-should-they-do-who-dont-get-ration-423998. [Online; posted 16-July-2016].

Patel, V. M., Ratha, N. K., & Chellappa, R. 2015. Cancelable Biometrics: A review. IEEE Signal Processing Magazine, **32**(5), 54–65.

Preneel, Bart, & Wyseur, Brecht. 2008. White-Box Cryptography.

PTI. 2015. Right to Privacy not a fundamental right, cannot be invoked to scrap Aadhar: Centre tells Supreme Court. http://articles.economictimes.indiatimes.com/2015-07-23/news/64773078_1_fundamental-right-attorney-general-mukul-rohatgi-privacy. [Online; posted 23-July-2015].

Sahai, Amit, & Waters, Brent. 2005. Fuzzy Identity-Based Encryption. Pages 457–473 of: EUROCRYPT.

Sakashita, Taiki, Shibata, Yoichi, Yamamoto, Takumi, Takahashi, Kenta, Ogata, Wakaha, Kikuchi, Hiroaki, & Nishigaki, Masakatsu. 2009. A Proposal of Efficient Remote Biometric Authentication Protocol. Berlin, Heidelberg: Springer Berlin Heidelberg. http://dx.doi.org/10.1007/978-3-642-04846-3_14. Pages 212–227.

Scroll Staff. 2016. Jaitley admits right to privacy but brazens it out on Money Bill manoeuvre for Aadhaar. http://scroll.in/article/805236/jaitley-admits-right-to-privacy-but-brazens-it-out-on-money-bill-manoeuvre-for-aadhar. [Online; posted 16-March-2016].

The London School of Economics and Political Science. 2005 (June). The Identity Project: An assessment of the UK Identity Cards Bill and its implications. http://www.lse.ac.uk/management/research/identityproject/identityreport.pdf.

The Planning Commission: Government of India. 2011 (December). Report of the Group of Experts on Privacy chaired by Justice A P Shah. http://planningcommission.nic.in/reports/genrep/rep_privacy.pdf.

Tulyakov, Sergey, Farooq, Faisal, & Govindaraju, Venu. 2005. Symmetric Hash Functions for Fingerprint Minutiae. Berlin, Heidelberg: Springer Berlin Heidelberg. Pages 30–38.

UIDAI. 2011. Aadhaar Security Policy & Framework for UIDAI Authentication (Version 1.0). http://uidai.gov.in/images/authDoc/d3_4_security_policy_framework_v1.pdf. [Online; accessed 31-July-2016].

UIDAI. 2014. AADHAAR TECHNOLOGY & ARCHITECTURE: Principles, Design, Best Practices, & Key Lessons. https://uidai.gov.in/images/AadhaarTechnologyArchitecture_March2014.pdf. [Online; accessed 31-July-2016].

UIDAI. 2016a. Authentication Overview. https://uidai.gov.in/auth.html. [Online; accessed 31-July-2016].

UIDAI. 2016b. Operating Model Overview. https://uidai.gov.in/authentication-2/operation-model.html. [Online; accessed 31-July-2016].

UN Global Pulse. 2012 (May). Big Data for Development: Challenges and Opportunities. http://www.unglobalpulse.org/sites/default/files/BigDataforDevelopment-UNGlobalPulseMay2012.pdf.

UN Global Pulse. 2016 (August). Data for Development: Challenges and Opportunities. http://www.unglobalpulse.org/. [Online; accessed 7-August-2016].

Varian, Hal R. 2014. Big Data: New Tricks for Econometrics. Journal of Economic Perspectives, **28**(2), 3–28. http://www.aeaweb.org/articles?id=10.1257/jep.28.2.3.

Vombatkere, Sudhir. 2016. How Aadhaar Neglects Personal Privacy and National Security. Mainstream, **LIV No 13**(March). http://www.mainstreamweekly.net/article6283.html. [Online; posted 15-August-2015].

Wang, Chenxi, Hill, Jonathan, Knight, John, & Davidson, Jack. 2000. Software Tamper Resistance: Obstructing Static Analysis of Programs. Tech. rept. Charlottesville, VA, USA.

Wikipedia. 2016a. Aadhaar. https://en.wikipedia.org/wiki/Aadhaar. [Online; accessed 31-July-2016].

Wikipedia. 2016b. Advanced Encryption Standard. https://en.wikipedia.org/wiki/Advanced_Encryption_Standard. [Online; accessed 30-July-2016].

Wikipedia. 2016c. Cryptographic hash function. https://en.wikipedia.org/wiki/Cryptographic_hash_function. [Online; accessed 30-July-2016].

Wikipedia. 2016d. Hardware security module. https://en.wikipedia.org/wiki/Hardware_security_module. [Online; accessed 30-July-2016].

Wikipedia. 2016e. Hash-based message authentication code. https://en.wikipedia.org/wiki/Hash-based_message_authentication_code. [Online; accessed 30-July-2016].

Wikipedia. 2016f. HTTPS. https://en.wikipedia.org/wiki/HTTPS. [Online; accessed 30-July-2016].

Wikipedia. 2016g. Kerberos (protocol). https://en.wikipedia.org/wiki/Kerberos_(protocol). [Online; accessed 30-July-2016].

Wikipedia. 2016h. Key management. https://en.wikipedia.org/wiki/Key_management. [Online; accessed 30-July-2016].

Wikipedia. 2016i. Man-in-the-middle attack. https://en.wikipedia.org/wiki/Man-in-the-middle_attack. [Online; accessed 30-July-2016].

Wikipedia. 2016j. Model Checking. https://en.wikipedia.org/wiki/Model_checking.

Wikipedia. 2016k. MPC. https://en.wikipedia.org/wiki/Secure_multi-party_computation.

Wikipedia. 2016l. OAuth. https://en.wikipedia.org/wiki/OAuth. [Online; accessed 30-July-2016].

Wikipedia. 2016m. Public key infrastructure. https://en.wikipedia.org/wiki/Public_key_infrastructure. [Online; accessed 30-July-2016].

Wikipedia. 2016n. Radius. https://en.wikipedia.org/wiki/RADIUS. [Online; accessed 30-July-2016].

Wikipedia. 2016o. Secure Hash Algorithm. https://en.wikipedia.org/wiki/Secure_Hash_Algorithm. [Online; accessed 30-July-2016].

Wikipedia. 2016p. Secure Shell. https://en.wikipedia.org/wiki/Secure_Shell. [Online; accessed 30-July-2016].

Wikipedia. 2016q. Shard (database architecture). https://en.wikipedia.org/wiki/Shard_(database_architecture). [Online; accessed 30-July-2016].

Wikipedia. 2016r. Side-channel attack. https://en.wikipedia.org/wiki/Side-channel_attack. [Online; accessed 31-July-2016].

Wikipedia. 2016s. Social Security Number. https://en.wikipedia.org/wiki/Social_Security_number. [Online; accessed 31-July-2016].

Wikipedia. 2016t. Static program analysis. https://en.wikipedia.org/wiki/Static_program_analysis.

Wikipedia. 2016u. Trusted Execution Technology. https://en.wikipedia.org/wiki/Trusted_Execution_Technology.

Wikipedia. 2016v. Virtual Private Network. https://en.wikipedia.org/wiki/Virtual_private_network. [Online; accessed 30-July-2016].

Wikipedia. 2016w. Zero-knowledge proof. https://en.wikipedia.org/wiki/Zero-knowledge_proof. [Online; accessed 30-July-2016].

Wyseur, Brecht, Michiels, Wil, Gorissen, Paul, & Preneel, Bart. 2007. Cryptanalysis of white-box DES implementations with arbitrary external encodings. Pages 264–277 of: International Workshop on Selected Areas in Cryptography. Springer.

Yadav, Anumeha. 2016a (April). Rajasthan presses on with Aadhaar after fingerprint readers fail: Well buy iris scanners. http://scroll.in/article/806243/rajasthan-presses-on-with-aadhaar-after-fingerprint-readers-fail-well-buy-iris-scanners. [Online; posted 10-April-2016].

Yadav, Anumeha. 2016b (August). Rajasthans living dead: Thousands of pensioners without Aadhaar or bank accounts struck off lists. http://scroll.in/article/813132/rajasthans-living-dead-thousands-of-pensioners-without-aadhaar-or-bank-accounts-struck-off-lists. [Online; posted 6-August-2016].

Zhong, Raymond. 2016. Is the Indian Government Saving as Much as It Says on Gas Subsidies? http://blogs.wsj.com/indiarealtime/2016/03/21/is-the-indian-government-saving-as-much-as-it-says-on-gas-subsidies/. [Online; posted 21-March-2016].