

Architecting for HIPAA Security and Compliance on Amazon Web Services

May 2017



© 2017, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Notices

This document is provided for informational purposes only. It represents AWS's current product offerings and practices as of the date of issue of this document, which are subject to change without notice. Customers are responsible for making their own independent assessment of the information in this document and any use of AWS's products or services, each of which is provided "as is" without warranty of any kind, whether express or implied. This document does not create any warranties, representations, contractual commitments, conditions or assurances from AWS, its affiliates, suppliers or licensors. The responsibilities and liabilities of AWS to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

Contents

Introduction	1
Encryption and Protection of PHI in AWS	2
Amazon EC2	2
Amazon Elastic Block Store	3
Amazon Redshift	4
Amazon S3	4
Amazon SQS	5
Amazon Glacier	6
Amazon RDS for MySQL	6
Amazon RDS for Oracle	7
Amazon RDS for PostgreSQL	8
Amazon Aurora	8
Elastic Load Balancing	9
Amazon EMR	10
Amazon DynamoDB	10
Amazon API Gateway	10
Using AWS KMS for Encryption of PHI	11
AWS Snowball	11
Auditing, Back-Ups, and Disaster Recovery	12
Document Revisions	13

Abstract

This paper briefly outlines how companies can use Amazon Web Services (AWS) to create HIPAA (Health Insurance Portability and Accountability Act)-compliant applications. We will focus on the HIPAA Privacy and Security Rules for protecting Protected Health Information (PHI), how to use AWS to encrypt data in transit and at rest, and how AWS features can be used to meet HIPAA requirements for auditing, back-ups, and disaster recovery.

Introduction

The Health Insurance Portability and Accountability Act of 1996 (HIPAA) applies to “covered entities” and “business associates.” Covered entities include health care providers engaged in certain electronic transactions, health plans, and health care clearinghouses. Business associates are entities that provide services to a covered entity that involve access by the business associate to Protected Health Information (PHI), as well as entities that create, receive, maintain, or transmit PHI on behalf of another business associate. HIPAA was expanded in 2009 by the Health Information Technology for Economic and Clinical Health (HITECH) Act. HIPAA and HITECH establish a set of federal standards intended to protect the security and privacy of PHI. HIPAA and HITECH impose requirements related to the use and disclosure of PHI, appropriate safeguards to protect PHI, individual rights, and administrative responsibilities. For additional information on HIPAA and HITECH, visit <http://www.hhs.gov/ocr/privacy/>.

Covered entities and their business associates can use the secure, scalable, low-cost IT components provided by Amazon Web Services (AWS) to architect applications in alignment with HIPAA and HITECH compliance requirements. AWS offers a commercial-off-the-shelf infrastructure platform with industry-recognized certifications and audits such as [ISO 27001](#), [FedRAMP](#), and the Service Organization Control Reports ([SOC1](#), [SOC2](#), and [SOC3](#)). AWS services and data centers have multiple layers of operational and physical security to help ensure the integrity and safety of customer data. With no minimum fees, no term-based contracts required, and pay-as-you-use pricing, AWS is a reliable and effective solution for growing health care industry applications.

AWS enables covered entities and their business associates subject to HIPAA to securely process, store, and transmit PHI. Additionally, AWS, as of July 2013, offers a standardized Business Associate Addendum (BAA) for such customers.

Customers who execute an AWS BAA may use any AWS service in an account designated as a HIPAA Account, but they may only process, store and transmit PHI using the HIPAA-eligible services defined in the AWS BAA. For a complete list of these services, see the [HIPAA Eligible Services Reference](#) page (<https://aws.amazon.com/compliance/hipaa-eligible-services-reference/>).

AWS maintains a standards-based risk management program to ensure that the HIPAA-eligible services specifically support the administrative, technical, and physical safeguards required under HIPAA. Using these services to store, process, and transmit PHI allows our customers and AWS to address the HIPAA requirements applicable to the AWS utility-based operating model.

Encryption and Protection of PHI in AWS

The HIPAA Security Rule includes addressable implementation specifications for the encryption of PHI in transmission (“in-transit”) and in storage (“at-rest”). Although this is an addressable implementation specification in HIPAA, AWS requires customers to encrypt PHI stored in or transmitted using HIPAA-eligible services in accordance with guidance from the Secretary of Health and Human Services (HHS), [Guidance to Render Unsecured Protected Health Information Unusable, Unreadable, or Indecipherable to Unauthorized Individuals](#). Please refer to this site because it may be updated, and may be made available on a successor (or related site) designated by HHS.

AWS offers a comprehensive set of features and services to make key management and encryption of PHI easy to manage and simpler to audit, including the AWS Key Management Service (AWS KMS). Customers with HIPAA compliance requirements have a great deal of flexibility in how they meet encryption requirements for PHI.

When determining how to implement encryption, customers may evaluate and take advantage of the encryption features native to the HIPAA-eligible services, or they can satisfy the encryption requirements through other means consistent with the guidance from HHS. The following sections provide high-level details about using available encryption features in each of the HIPAA-eligible services and other patterns for encrypting PHI, and how AWS KMS can be used to encrypt the keys used for encryption of PHI on AWS.

Amazon EC2

Amazon EC2 is a scalable, user-configurable compute service that supports multiple methods for encrypting data at rest. For example, customers might elect to perform application- or field-level encryption of PHI as it is processed within an application or database platform hosted in an Amazon EC2 instance. Approaches range from encrypting data using standard libraries in an

application framework such as Java or .NET; leveraging Transparent Data Encryption features in Microsoft SQL or Oracle; or by integrating other third-party and software as a service (SaaS)-based solutions into their applications. Customers can choose to integrate their applications running in Amazon EC2 with AWS KMS SDKs, simplifying the process of key management and storage. Customers can also implement encryption of data at rest using file-level or full disk encryption (FDE) by utilizing third-party software from [AWS Marketplace Partners](#) or native file system encryption tools (such as dm-crypt, LUKS, etc.).

Network traffic containing PHI must encrypt data in transit. For traffic between external sources (such as the Internet or a traditional IT environment) and Amazon EC2, customers should use industry-standard transport encryption mechanisms such as TLS or IPsec virtual private networks (VPNs), consistent with the [Guidance](#). Internal to an Amazon Virtual Private Cloud (VPC) for data traveling between Amazon EC2 instances, network traffic containing PHI must also be encrypted; most applications support TLS or other protocols providing in-transit encryption that can be configured to be consistent with the Guidance. For applications and protocols that do not support encryption, sessions transmitting PHI can be sent through encrypted tunnels using IPsec or similar implementations between instances.

Amazon Virtual Private Cloud offers a set of network security features well-aligned to architecting for HIPAA compliance. Features such as stateless network access control lists and dynamic reassignment of instances into stateful security groups afford flexibility in protecting the instances from unauthorized network access. Amazon VPC also allows customers to extend their own network address space into AWS, as well as providing a number of ways to connect their data centers to AWS. VPC Flow Logs provide an audit trail of accepted and rejected connections to instances processing, transmitting or storing PHI. For more information on Amazon VPC, see <http://aws.amazon.com/vpc/>.

Amazon Elastic Block Store

Amazon EBS encryption at rest is consistent with the Guidance that is in effect at the time of publication of this whitepaper. Because the Guidance might be updated, customers should continue to evaluate and determine whether Amazon EBS encryption satisfies their compliance and regulatory requirements. With Amazon EBS encryption, a unique volume encryption key is generated for

each EBS volume; customers have the flexibility to choose which master key from the AWS Key Management Service is used to encrypt each volume key. For more information, see

<http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/EBSEncryption.html>

Amazon Redshift

Amazon Redshift provides database encryption for its clusters to help protect data at rest. When customers enable encryption for a cluster, Amazon Redshift encrypts all data, including backups, by using hardware-accelerated Advanced Encryption Standard (AES)-256 symmetric keys. Amazon Redshift uses a four-tier, key-based architecture for encryption. These keys consist of data encryption keys, a database key, a cluster key, and a master key. The cluster key encrypts the database key for the Amazon Redshift cluster. Customers can use either AWS KMS or an AWS CloudHSM (Hardware Security Module) to manage the cluster key. Amazon Redshift encryption at rest is consistent with the Guidance that is in effect at the time of publication of this whitepaper. Because the Guidance might be updated, customers should continue to evaluate and determine whether Amazon Redshift encryption satisfies their compliance and regulatory requirements. For more information see

<http://docs.aws.amazon.com/redshift/latest/mgmt/working-with-db-encryption.html>.

Connections to Amazon Redshift containing PHI must use transport encryption and customers should evaluate the configuration for consistency with the Guidance. For more information, see

<http://docs.aws.amazon.com/redshift/latest/mgmt/connecting-ssl-support.html>.

Amazon S3

Customers have several options for encryption of data at rest when using Amazon S3, including both server-side and client-side encryption and several methods of managing keys. For more information see

<http://docs.aws.amazon.com/AmazonS3/latest/dev/UsingEncryption.html>.

Connections to Amazon S3 containing PHI must use endpoints that accept encrypted transport (HTTPS). For a list of regional endpoints, see http://docs.aws.amazon.com/general/latest/gr/rande.html#s3_region.

Customers should not use PHI in bucket names, object names, or metadata because this data is not encrypted using S3 server-side encryption and is not generally encrypted in client-side encryption architectures.

Amazon SQS

Customers should understand the following key encryption requirements in order to use Amazon SQS with Protected Health Information (PHI).

- Communication with the Amazon SQS Queue via the Query Request must be encrypted with HTTPS. For more information on making SQS requests, see http://docs.aws.amazon.com/AWSSimpleQueueService/latest/SQSDeveloperGuide/MakingRequests_MakingQueryRequestsArticle.html.
- Amazon SQS supports server-side encryption integrated with the AWS Key Management Service (KMS) to protect data at rest. The addition of server-side encryption allows you to transmit and receive sensitive data with the increased security of using encrypted queues. Amazon SQS server-side encryption uses the 256-bit Advanced Encryption Standard (AES-256 GCM algorithm) to encrypt the body of each message. The integration with KMS allows you to centrally manage the keys that protect Amazon SQS messages along with keys that protect your other AWS resources. AWS KMS logs every use of your encryption keys to AWS CloudTrail to help meet your regulatory and compliance needs. For more information, and to check your region for the availability for SSE for Amazon SQS, see: <http://docs.aws.amazon.com/AWSSimpleQueueService/latest/SQSDeveloperGuide/sqs-server-side-encryption.html>.
- If server-side encryption is not used, the message payload itself must be encrypted prior to being sent to SQS. One way to encrypt the message payload is by using the Amazon SQS Extended Client along with the Amazon S3 encryption client. For more information on using client-side encryption, see <https://aws.amazon.com/blogs/developer/encrypting->

[message-payloads-using-the-amazon-sqs-extended-client-and-the-amazon-s3-encryption-client/](#).

Amazon SQS is integrated with CloudTrail, a service that logs API calls made by or on behalf of Amazon SQS in your AWS account and delivers the log files to the specified Amazon S3 bucket. CloudTrail captures API calls made from the Amazon SQS console or from the Amazon SQS API. You can use the information collected by CloudTrail to determine which requests are made to Amazon SQS, the source IP address from which the request is made, who made the request, when it is made, and so on. For more information on logging SQS operations, see

<http://docs.aws.amazon.com/AWSSimpleQueueService/latest/SQSDeveloperGuide/logging-using-cloudtrail.html>.

Amazon Glacier

Amazon Glacier automatically encrypts data at rest using AES 256-bit symmetric keys and supports secure transfer of customer data over secure protocols.

Connections to Amazon Glacier containing PHI must use endpoints that accept encrypted transport (HTTPS). For a list of regional endpoints, see http://docs.aws.amazon.com/general/latest/gr/rande.html#glacier_region.

Customers should not use PHI in archive and vault names or metadata because this data is not encrypted using Amazon Glacier server-side encryption and is not generally encrypted in client-side encryption architectures.

Amazon RDS for MySQL

Amazon RDS for MySQL allows customers to encrypt MySQL databases using keys that customers manage through AWS KMS. On a database instance running with Amazon RDS encryption, data stored at rest in the underlying storage is encrypted consistent with the Guidance in effect at the time of publication of this whitepaper, as are automated backups, read replicas, and snapshots. Because the Guidance might be updated, customers should continue to evaluate and determine whether Amazon RDS for MySQL encryption satisfies their compliance and regulatory requirements. For more information on encryption at rest using Amazon RDS, see

<http://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Overview.Encryption.html>.

Connections to RDS for MySQL containing PHI must use transport encryption. For more information on enabling encrypted connections, see <http://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/UsingWithRDS.SSL.html>.

Amazon RDS for Oracle

Customers have several options for encrypting PHI at rest using Amazon RDS for Oracle.

Customers can encrypt Oracle databases using keys that customers manage through AWS KMS. On a database instance running with Amazon RDS encryption, data stored at rest in the underlying storage is encrypted consistent with the Guidance in effect at the time of publication of this whitepaper, as are automated backups, read replicas, and snapshots. Because the Guidance might be updated, customers should continue to evaluate and determine whether Amazon RDS for Oracle encryption satisfies their compliance and regulatory requirements. For more information on encryption at-rest using Amazon RDS, see <http://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Overview.Encryption.html>.

Customers can also leverage Oracle Transparent Data Encryption (TDE), and customers should evaluate the configuration for consistency with the Guidance. Oracle TDE is a feature of the Oracle Advanced Security option available in Oracle Enterprise Edition. This feature automatically encrypts data before it is written to storage and automatically decrypts data when the data is read from storage. Customers can also use AWS CloudHSM to store Amazon RDS Oracle TDE keys. For more information, see the following:

- Amazon RDS for Oracle Transparent Data Encryption:
<http://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Appendix.Oracle.Options.AdvSecurity.html>.
- Using AWS CloudHSM to store Amazon RDS Oracle TDE keys:
<http://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Appendix.OracleCloudHSM.html>.

Connections to Amazon RDS for Oracle containing PHI must use transport encryption and evaluate the configuration for consistency with the Guidance. This is accomplished using Oracle Native Network Encryption and enabled in Amazon RDS for Oracle option groups. For detailed information, see <http://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Appendix.Oracle.Options.NetworkEncryption.html>.

Amazon RDS for PostgreSQL

Amazon RDS for PostgreSQL allows customers to encrypt PostgreSQL databases using keys that customers manage through AWS KMS. On a database instance running with Amazon RDS encryption, data stored at rest in the underlying storage is encrypted consistent with the Guidance in effect at the time of publication of this whitepaper, as are automated backups, read replicas, and snapshots. Because the Guidance might be updated, customers should continue to evaluate and determine whether Amazon RDS for PostgreSQL encryption satisfies their compliance and regulatory requirements. For more information on encryption at rest using Amazon RDS, see <http://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Overview.Encryption.html>.

Connections to RDS for PostgreSQL containing PHI must use transport encryption. For more information on enabling encrypted connections, see <http://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/UsingWithRDS.SSL.html>.

Amazon Aurora

Amazon Aurora allows customers to encrypt Aurora databases using keys that customers manage through AWS KMS. On a database instance running with Amazon Aurora encryption, data stored at rest in the underlying storage is encrypted consistent with the Guidance in effect at the time of publication of this whitepaper, as are automated backups, read replicas, and snapshots. Because the Guidance might be updated, customers should continue to evaluate and determine whether Amazon Aurora encryption satisfies their compliance and regulatory requirements. For more information on encryption at rest using Amazon RDS, see <http://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Overview.Encryption.html>.

Currently, customers may only use the MySQL-compatible edition of Amazon Aurora as part of our BAA.

Connections to Aurora containing PHI must use transport encryption. For more information on enabling encrypted connections, see <http://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/UsingWithRDS.SSL.html>.

Elastic Load Balancing

Customers may use Elastic Load Balancing to terminate and process sessions containing PHI. Customers may choose either the Classic Load balancer or the Application Load Balancer. Because all network traffic containing PHI must be encrypted in transit end-to-end, customers have the flexibility to implement two different architectures:

Customers can terminate HTTPS, HTTP/2 over TLS (for Application), or SSL/TLS on Elastic Load Balancing by creating a load balancer that uses an encrypted protocol for connections. This feature enables traffic encryption between the customer's load balancer and the clients that initiate HTTPS, HTTP/2 over TLS, or SSL/TLS sessions, and for connections between the load balancer and customer back-end instances. Sessions containing PHI must encrypt both front-end and back-end listeners for transport encryption. Customers should evaluate their certificates and session negotiation policies and maintain them consistent to the Guidance. For more information, see <http://docs.aws.amazon.com/ElasticLoadBalancing/latest/DeveloperGuide/elb-https-load-balancers.html>.

Alternatively, customers can configure Amazon ELB in basic TCP-mode (for Classic) or over WebSockets (for Application) and pass-through encrypted sessions to back-end instances where the encrypted session is terminated. In this architecture, customers manage their own certificates and TLS negotiation policies in applications running in their own instances. For more information, see <http://docs.aws.amazon.com/ElasticLoadBalancing/latest/DeveloperGuide/elb-listener-config.html>.

In both architectures, customers should implement a level of logging which they determine to be consistent with HIPAA and HITECH requirements.

Amazon EMR

Amazon EMR deploys and manages a cluster of Amazon EC2 instances into a customer's account.

For information on encryption with Amazon EMR, please see <https://docs.aws.amazon.com/ElasticMapReduce/latest/ReleaseGuide/emr-data-encryption-options.html>.

Amazon DynamoDB

Connections to Amazon DynamoDB containing PHI must use endpoints that accept encrypted transport (HTTPS). For a list of regional endpoints, see http://docs.aws.amazon.com/general/latest/gr/rande.html#ddb_region.

PHI stored in Amazon DynamoDB must be encrypted at-rest consistent with the Guidance. Amazon DynamoDB customers can use the application development framework of their choice to encrypt PHI in applications before storing the data in Amazon DynamoDB. Alternatively, a client-side library for encrypting content is available from the AWS Labs GitHub repository. Customers may evaluate this implementation for consistency with the Guidance. For more information, see <https://github.com/aws-labs/aws-dynamodb-encryption-java>. Careful consideration should be taken when selecting primary keys and when creating indexes such that unsecured PHI is not required for queries and scans in Amazon DynamoDB.

Amazon API Gateway

Customers may use Amazon API Gateway to process and transmit PHI. While Amazon API Gateway automatically uses HTTPS endpoints for encryption in-flight, customers may also choose to encrypt payloads client-side. API Gateway passes all non-cached data through memory and does not write it to disk. Customers may use AWS Signature Version 4 for authorization with API Gateway. For more information, see the following:

- <https://aws.amazon.com/api-gateway/faqs/#security>
- <https://docs.aws.amazon.com/apigateway/latest/developerguide/apigateway-control-access-to-api.html>

Customers may integrate with any service that is connected to API Gateway, provided that when PHI is involved, the service is configured consistent with the Guidance and BAA. For information on integrating API Gateway with back end services, see

<https://docs.aws.amazon.com/apigateway/latest/developerguide/how-to-method-settings.html>.

Customers can use AWS CloudTrail and Amazon CloudWatch to enable logging that is consistent with their logging requirements. Customers should ensure that any PHI sent through API Gateway (such as in headers, URLs, and request/response) is only captured by HIPAA-eligible services that have been configured to be consistent with the Guidance. For more information on logging with API Gateway, see <https://aws.amazon.com/premiumsupport/knowledge-center/api-gateway-cloudwatch-logs/>.

Using AWS KMS for Encryption of PHI

Master keys in AWS KMS can be used to encrypt/decrypt data encryption keys used to encrypt PHI in customer applications or in AWS services that are integrated with AWS KMS. AWS KMS can be used in conjunction with a HIPAA account, but PHI may only be processed, stored, or transmitted in HIPAA-eligible services. KMS does not need to be a HIPAA-eligible service so long as it is used to generate and manage keys for applications running in other HIPAA-eligible services. For example, an application processing PHI in Amazon EC2 could use the GenerateDataKey API call to generate data encryption keys for encrypting and decrypting PHI in the application. The data encryption keys would be protected by customer master keys stored in AWS KMS, creating a highly auditable key hierarchy as API calls to AWS KMS are logged in AWS CloudTrail.

AWS Snowball

With AWS Snowball (Snowball), you can transfer hundreds of terabytes or petabytes of data between your on-premises data centers and Amazon Simple Storage Service (Amazon S3).

PHI stored in AWS Snowball must be encrypted at-rest consistent with the Guidance. When creating an import job, customers will need to specify the ARN for the AWS Key Management Service (AWS KMS) master key to be used to

protect data within the Snowball. In addition, during the creation of the import job, customers should choose a destination S3 bucket that meets the encryption standards set by the Guidance. While Snowball does not currently support server-side encryption with AWS KMS-managed keys (SSE-KMS) or server-side encryption with customer provided keys (SSE-C), Snowball does support server-side encryption with Amazon S3-managed encryption keys (SSE-S3). For more information see [Protecting Data Using Server-Side Encryption with Amazon S3-Managed Encryption Keys \(SSE-S3\)](#).

Alternatively, customers can use the encryption methodology of their choice to encrypt PHI before storing the data in AWS Snowball.

Currently, customers may only use the standard AWS Snowball appliance as part of our BAA.

Auditing, Back-Ups, and Disaster Recovery

HIPAA's Security Rule also requires in-depth auditing capabilities, data back-up procedures, and disaster recovery mechanisms. The services in AWS contain many features that help customers address these requirements.

In designing an information system that is consistent with HIPAA and HITECH requirements, customers should put auditing capabilities in place to allow security analysts to examine detailed activity logs or reports to see who had access, IP address entry, what data was accessed, etc. This data should be tracked, logged, and stored in a central location for extended periods of time, in case of an audit. Using Amazon EC2, customers can run activity log files and audits down to the packet layer on their virtual servers, just as they do on traditional hardware. They also can track any IP traffic that reaches their virtual server instance. A customer's administrators can back up the log files into Amazon S3 for long-term reliable storage.

Under HIPAA, covered entities must have a contingency plan to protect data in case of an emergency and must create and maintain retrievable exact copies of electronic PHI. To implement a data back-up plan on AWS, Amazon EBS offers persistent storage for Amazon EC2 virtual server instances. These volumes can be exposed as standard block devices, and they offer off-instance storage that persists independently from the life of an instance. To align with HIPAA guidelines, customers can create point-in-time snapshots of Amazon EBS

volumes that automatically are stored in Amazon S3 and are replicated across multiple Availability Zones, which are distinct locations engineered to be insulated from failures in other Availability Zones. These snapshots can be accessed at any time and can protect data for long-term durability. Amazon S3 also provides a highly available solution for data storage and automated back-ups. By simply loading a file or image into Amazon S3, multiple redundant copies are automatically created and stored in separate data centers. These files can be accessed at any time, from anywhere (based on permissions), and are stored until intentionally deleted.

Disaster recovery, the process of protecting an organization's data and IT infrastructure in times of disaster, is typically one of the more expensive HIPAA requirements to comply with. This involves maintaining highly available systems, keeping both the data and system replicated off-site, and enabling continuous access to both. AWS inherently offers a variety of disaster recovery mechanisms.

With Amazon EC2, administrators can start server instances very quickly and can use an Elastic IP address (a static IP address for the cloud computing environment) for graceful failover from one machine to another. Amazon EC2 also offers Availability Zones. Administrators can launch Amazon EC2 instances in multiple Availability Zones to create geographically diverse, fault tolerant systems that are highly resilient in the event of network failures, natural disasters, and most other probable sources of downtime. Using Amazon S3, a customer's data is replicated and automatically stored in separate data centers to provide reliable data storage designed to provide 99.99% availability.

For more information on disaster recovery, see the AWS Disaster Recovery whitepaper available at <http://aws.amazon.com/disaster-recovery/>.

Document Revisions

Date	Description
May 2017	Removed requirement for Dedicated Instances or Dedicated Hosts for processing PHI in EC2 and EMR.
March 2017	Updated list of services to point to AWS Services in Scope by Compliance Program page. Added description for Amazon API Gateway.

Date	Description
January 2017	Updated to newest template.
October 2016	First publication