



UNIPAY DEPLOYMENT CONFIGURATIONS



Introduction	3
Deployment Configuration.....	3
Deployment Configuration Diagram	4
Inventory Parameters Diagram.....	5
Minimal System Requirements* (for production version of UniPay application).....	6
Minimal System Requirements** (for test version of UniPay application)	6
Required Middleware Products	7
Keepalived.....	7
HAProxy	7
ProFTPD	7
JBoss	7
MySQL.....	8
Redis	8
Elastic Stack	8
Jenkins	9
JavaMelody	9
StrongAuth (Tokenization Appliance)	9
SafeNet HSM.....	9
Migration Requirements	10



Introduction

The purpose of this document is to define deployment configuration of the UniPay application.

Below you can find detailed deployment configuration diagram, as well as inventory diagram, minimal system requirements and brief description of middleware products, required to have UniPay running.

Deployment Configuration

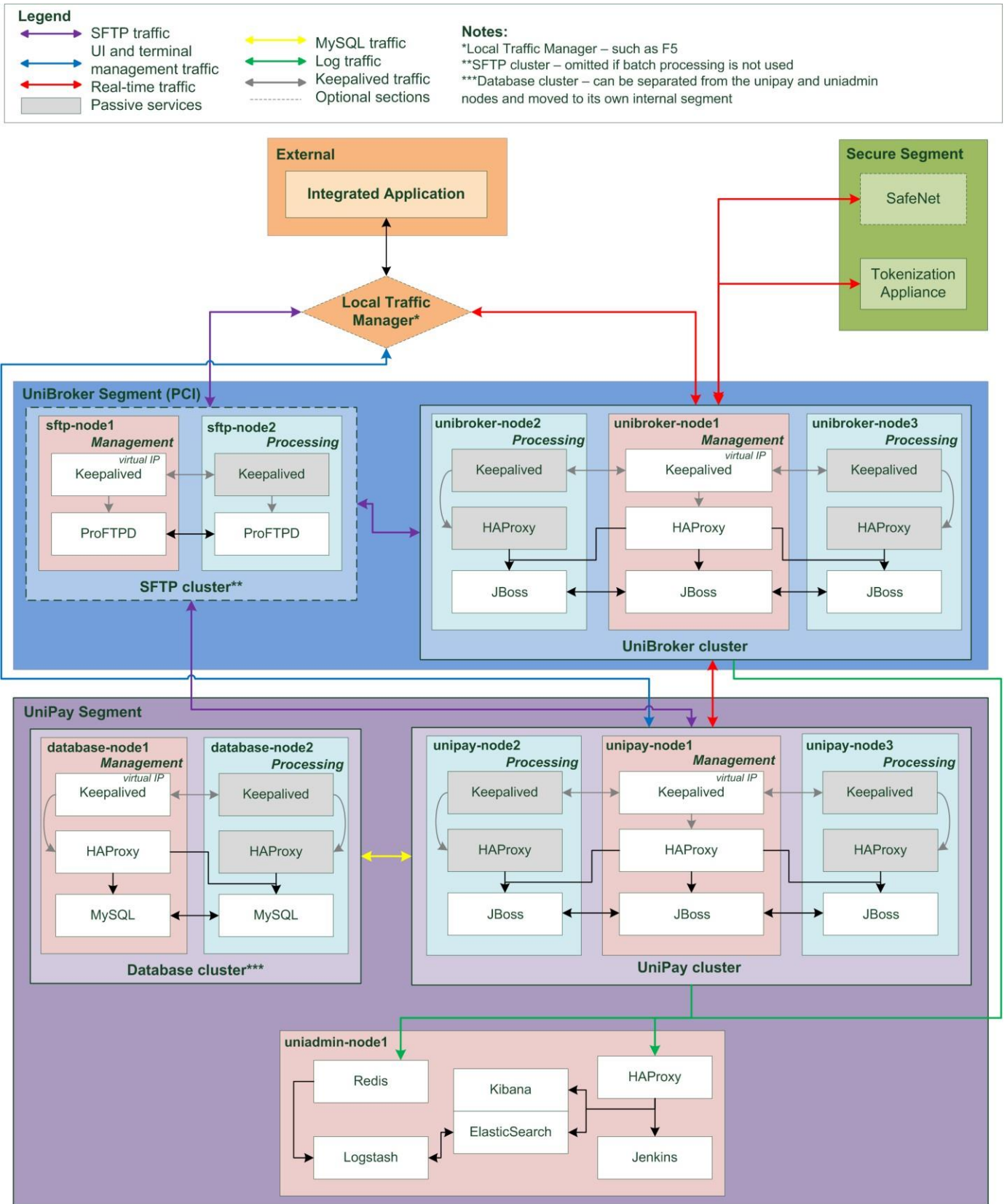
Deployment configuration may consist of the following elements:

- UniPay (required) – the application itself;
- Database (required) – data storage;
- Tokenization appliance (required) – sensitive card data storage;
- UniBroker (required) – proxy-filter used for data sanitization; simplifies PCI audit, and makes the whole processing more secure.
- SFTP server (optional) – server used for exchanging of files when batch processing is used;
- HSM (optional) – appliance used for decryption of card data when point-to-point encryption is used;
- Admin node (required) – a collection of administrative tools for deployment and update of the application as well as for log storage.

UniPay has clustered deployment configuration, which has higher availability and higher transaction throughput as compared to deprecated non-clustered deployment configuration.

To manage UniPay and UniBroker instances, admin node is used. It consists of the Jenkins application, which contains the tasks that simplify deployment and update of the application, as well as the ElasticStack and accompanying tools, necessary for collection of logs across the system in a centralized storage for subsequent centralized access.

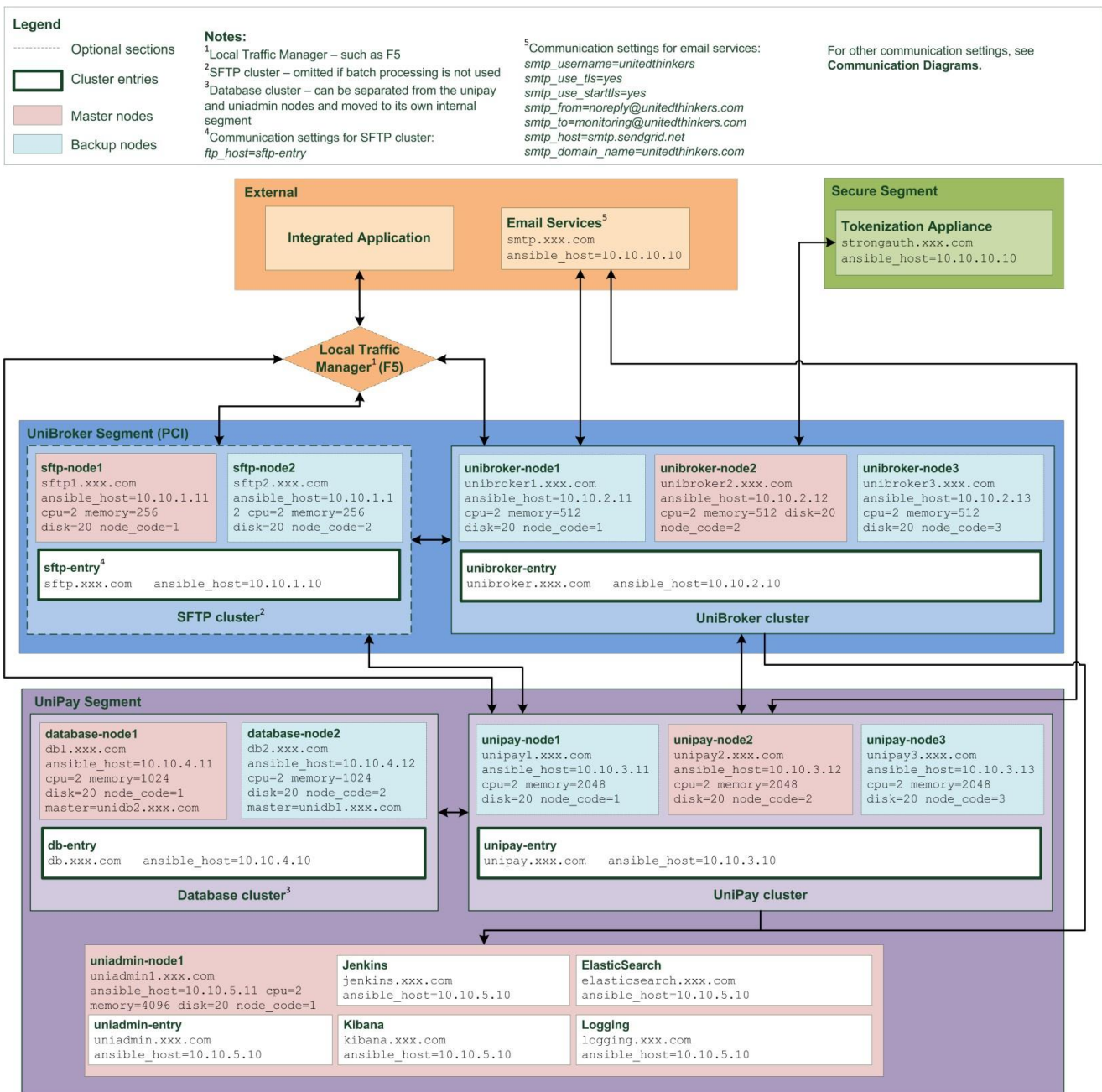
Deployment Configuration Diagram



Inventory Parameters Diagram

Before deployment of the UniPay application, you have to review the configuration diagram above and fill out the [questionnaire](#), sent by the gateway support team. The questionnaire collects the information about your domain names, IP addresses, etc. See the [questionnaire explanation](#) for more information on how to fill out the questionnaire document in a proper way.

The data from the questionnaire is filled out in the Inventory properties file, which is used during build process of the UniPay application. The Inventory file defines the configuration of your system (for example, which clusters to create, how many nodes to include in each cluster, etc). See [Deployment Management Guide](#) for more details about UniPay deployment and management of the Inventory file. The Inventory diagram below shows how the parameters from your Inventory file correspond to the elements of UniPay configuration.



Minimal System Requirements*
(for production version of UniPay application)

Node Name	Recommended CPU, Cores	Recommended RAM, GB	Recommended HDD, GB	Recommended OS
sftp-node1	2	1-2	80	Oracle Linux 8 (recommended) Red Hat Linux 8
sftp-node2	2	1-2	80	
unibroker-node1	2	6-8	80	
unibroker-node2	2	6-8	80	
unibroker-node3	2	6-8	80	
unipay-node1	4-8	8-14	100	
unipay-node2	4-8	8-14	100	
unipay-node3	4-8	8-14	100	
mysql-node1	16	32	500	
mysql-node2	16	32	500	
admin-node	4	4	1050	

Minimal System Requirements**
(for test version of UniPay application)

Node Name	Recommended CPU, Cores	Recommended RAM, GB	Recommended HDD, GB	Recommended OS
sftp-node1	2	1	40	Oracle Linux 8 (recommended) Red Hat Linux 8
sftp-node2	2	1	40	
unibroker-node1	2	2	40	
unibroker-node2	2	2	40	
unibroker-node3	2	2	40	
unipay-node1	2	6	50	
unipay-node2	2	6	50	
unipay-node3	2	6	50	
mysql-node1	4	4	200	
mysql-node2	4	4	200	
admin-node	4	4	400	

*Note that these are minimal system requirements for the gateway only. These requirements do not take into account resources consumed by operating system, antivirus, monitoring software and other applications you are planning to run. To be able to install and use additional applications, recommended RAM and HDD must be increased by amount consumed by these applications. For example, if your applications consume 4GB of RAM, recommended RAM must be increased by 4GB respectively.

**Given the requirements of development and PCI, production and test environments should use the same configuration, although the amount of the resources and number of nodes can be decreased for sandbox environment. For example, if the production server includes 4 cores and 32GB of RAM, test server can operate using just 2 cores and 8GB of RAM.

Required Middleware Products

Keepalived

<http://www.keepalived.org/>

Keepalived is a routing software written in C. The main goal of this project is to provide simple and robust facilities for loadbalancing and high-availability to Linux system and Linux based infrastructures. Loadbalancing framework relies on well-known and widely used Linux Virtual Server (IPVS) kernel module providing Layer4 loadbalancing. Keepalived implements a set of checkers to dynamically and adaptively maintain and manage loadbalanced server pool according their health. On the other hand high-availability is achieved by VRRP protocol. In addition, Keepalived implements a set of hooks to the VRRP finite state machine providing low-level and high-speed protocol interactions. Keepalived frameworks can be used independently or all together to provide resilient infrastructures.

Recommended version: 1.3.5.

HAProxy

<http://www.haproxy.org/>

HAProxy is a free, very fast and reliable solution offering high availability, load balancing, and proxying for TCP and HTTP-based applications. It is particularly suited for very high traffic web sites and powers quite a number of the world's most visited ones. Over the years it has become the de-facto standard opensource load balancer, is now shipped with most mainstream Linux distributions, and is often deployed by default in cloud platforms.

Recommended version: 1.5.18.

ProFTPD

<http://www.proftpd.org/>

ProFTPD is a highly configurable GPL-licensed FTP server software. ProFTPD is free and open-source software, compatible with Unix-like systems and Microsoft Windows (via Cygwin). Along with vsftpd and Pure-FTPd, ProFTPD is among the most popular FTP servers in Unix-like environments today. Compared to those, which focus e.g. on simplicity, speed or security, ProFTPD's primary design goal is to be a highly feature rich FTP server, exposing a large amount of configuration options to the user.

Recommended version: 1.3.5a.

JBoss

<http://www.jboss.org/>

The JBoss Enterprise Application Platform (or JBoss EAP) is a subscription-based/open-source Java EE-based application server runtime platform used for building, deploying, and hosting highly-transactional Java applications and services. The JBoss Enterprise Application Platform is part of the JBoss Enterprise Middleware portfolio of software. Because it is Java-based, the JBoss application server operates across platforms; it is usable on any operating system that supports Java.

Recommended version: EAP 7.3

MySQL

<https://www.mysql.com/>

MySQL is an open-source relational database management system (RDBMS) that uses Structured Query Language (SQL). MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP open-source web application software stack. Free-software open-source projects that require a full-featured database management system often use MySQL.

Recommended version: 5.7.33.

Redis

<http://redis.io/>

Redis is an open source (BSD licensed), in-memory data structure store, used as database, cache and message broker. It supports data structures such as strings, hashes, lists, sets, sorted sets with range queries, bitmaps, hyperloglogs and geospatial indexes with radius queries. Redis has built-in replication, Lua scripting, LRU eviction, transactions and different levels of on-disk persistence, and provides high availability via Redis Sentinel and automatic partitioning with Redis Cluster.

Recommended version: 3.2.12.

Elastic Stack

<https://www.elastic.co/>

ElasticStack is used for log storage and subsequent logs analysis. It consists of the following programs: **Elasticsearch**, **Logstash**, and **Kibana**.

- **Elasticsearch** is a distributed, open source search and analytics engine, designed for horizontal scalability, reliability, and easy management. It combines the speed of search with the power of analytics via a sophisticated, developer-friendly query language covering structured, unstructured, and time-series data.

Recommended version: 2.4.6.

- **Logstash** is a flexible, open source data collection, enrichment, and transportation pipeline. With connectors to common infrastructure for easy integration, Logstash is designed to efficiently process a growing list of log, event, and unstructured data sources for distribution into a variety of outputs, including Elasticsearch.

Recommended version: 2.4.1.

- **Kibana** is an open source data visualization platform that allows you to interact with your data through stunning, powerful graphics. From histograms to geomaps, Kibana brings your data to life with visuals that can be combined into custom dashboards that help you share insights from your data far and wide.

Recommended version: 4.5.4.

Jenkins

<https://jenkins.io>

Jenkins is an open source continuous integration tool written in Java. Jenkins provides continuous integration services for software development. It is a server-based system running in a servlet container such as Apache Tomcat. It supports SCM tools including AccuRev, CVS, Subversion, Git, Mercurial, Perforce, Clearcase and RTC, and can execute Apache Ant and Apache Maven based projects as well as arbitrary shell scripts and Windows batch commands.

Jenkins is used for administration of all nodes within clustered environment. It allows a system administrator to rebuild and deploy the application, manage services on the nodes, synchronize resources between the nodes, etc. To learn more about administrative scripts available within Jenkins instance, follow the [link](#).

Recommended version: 2.204.1.

JavaMelody

<https://github.com/javamelody/javamelody/wiki>

JavaMelody is an opensource tool, which is easy to integrate and is lightweight (no profiling and no database). The goal of JavaMelody is to monitor Java or Java EE applications in QA and production environments. It is a tool to measure and calculate statistics on real operation of an application depending on the usage of the application by users. JavaMelody is mainly based on statistics of requests and on evolution charts.

Recommended version: 1.55.0.

StrongAuth (Tokenization Appliance)

<http://www.strongauth.com/>

StrongAuth helps protect data with strong encryption, so that even if a company's network infrastructure is breached, its critical data – including customers' credit card numbers, for example – is still safe. Combining encryption, tokenization, strong-authentication, cryptographic module, high-availability and an open-source licensing model, the latest edition of StrongAuth's flagship solution provides data-protection unlike any other product on the market. StrongAuth software is open source.

SafeNet HSM

<https://safenet.gemalto.com/data-encryption/hardware-security-modules-hsms/safenet-payment-hsm/>

SafeNet Payment HSM – formerly Luna EFT – provides FIPS 140-2 Level 3 certified physical and logical protection to cryptographic keys use to secure financial transactions. As a PCI-Certified hardware security module (HSM), SafeNet Payment HSM adheres to the highest level of security in the industry. The PCI HSM standard, first issued by the Payment Card Industry Security Standards Council (PCI SSC) in 2009, defines a set of requirements for HSMs to use in several areas of the payment process.



Migration Requirements

During UniPay operation, there may be a need to migrate from one data center to another. In that case, the second datacenter is required to have the same configuration as the first one. Any changes to the datacenter can be applied only after the migration process is finished.