Yeti Project Phase-2

Draft

2020

Decentralized Distributed DM System

BII Lab

Term

Directory	Description
IANA	The Internet Assigned Numbers Authority
DM	Distribution Master
DNS	Domain Name System
TLD	Top Level Domain
RS	Root Server
P2P	Peer to Peer
HSM	Hardware Security Module
3PC	Three-phase Commit
DNSSEC	Domain Name System Security Extensions
SOA	Start of Authority
DNSKEY	Domain Name System KEY
RRSIG	Resource Record Signature
NS	Name Server
RRset	A set of resource records
KSK	Key signing Key
ZSK	Zone signing Key
XFR	Zone Transfer
PBFT	Practical Byzantine Fault Tolerance
DKG	Distributed Key Generation

Term

Catalogue	Description
QDM	Quit DM
UDM	Update DM
DMMC	DM Management Committee
FCFS	First Come First Served
TS	Threshold Signature
PoWE	Proof-of-Work-Efficiency

CONTENTS

Project Overview 01

- 1.1 Background
- 1.2 Project Overview
- 1.3 DMMC Introduction
- 1.4 Network Topology and Algorithms
- 1.5 Architecture
  1.6 Running Procedure Example

DM Network 02

- 2.1 Network Initialization
- 2.2 Election
- 2.3 Reward and Punishment
- 2.4 Heartbeat
- 2.5 DM Joining Process
- 2.6 DM Withdraw Process
- 2.7 DM Kick-out Process
- 2.8 DM Update Process

Zone File Generation 03

- 3.1 Acquisition of Zone File
- 3.2 Zone File Generation Specification
- 3.3 Consensus on Zone File
- 3.4 File Synchronization

04 Threshold Signature

- 4.1 HSM
- 4.2 Threshold Signature
- 4.4 Key Management

05 Signed Root Zone File Synchronization

- 5.1 Signed Root Zone File Synchronization

06 DM and Root Server Synchronization

- 6.1 Root Server and DM Synchronization Scheme

07 DM Security Design

- 7.1 Security Consideration

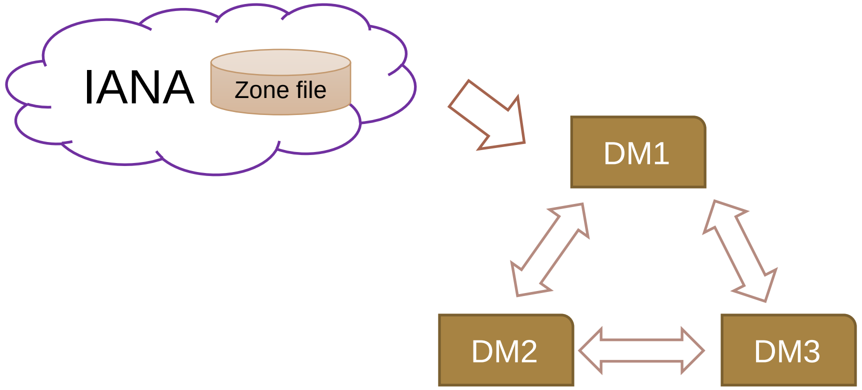
/01 Project Overview

1.1 Background

Yeti Project Phase-1 adopts 3DM scheme to enhance system redundancy and DM management mechanism, but there is still room for optimization：

1. Decentralization: 3 DM management rights are relatively independent, and decentralization is not complete enough
2. Message is too long: 3 DM uses multiple DNSKEY, DNSKEY redundancy to increase the length of the response message
3. Others ： 3 DM may cause zone file to fork

Project details ： https://github.com/BII-Lab/yeti-Project/blob/master/doc/yeti-DM-Sync-MZSK.md
https://yeti-dns.org/yeti/blog/2018/08/13/fault-tolerant-distribution-master-architecture.html



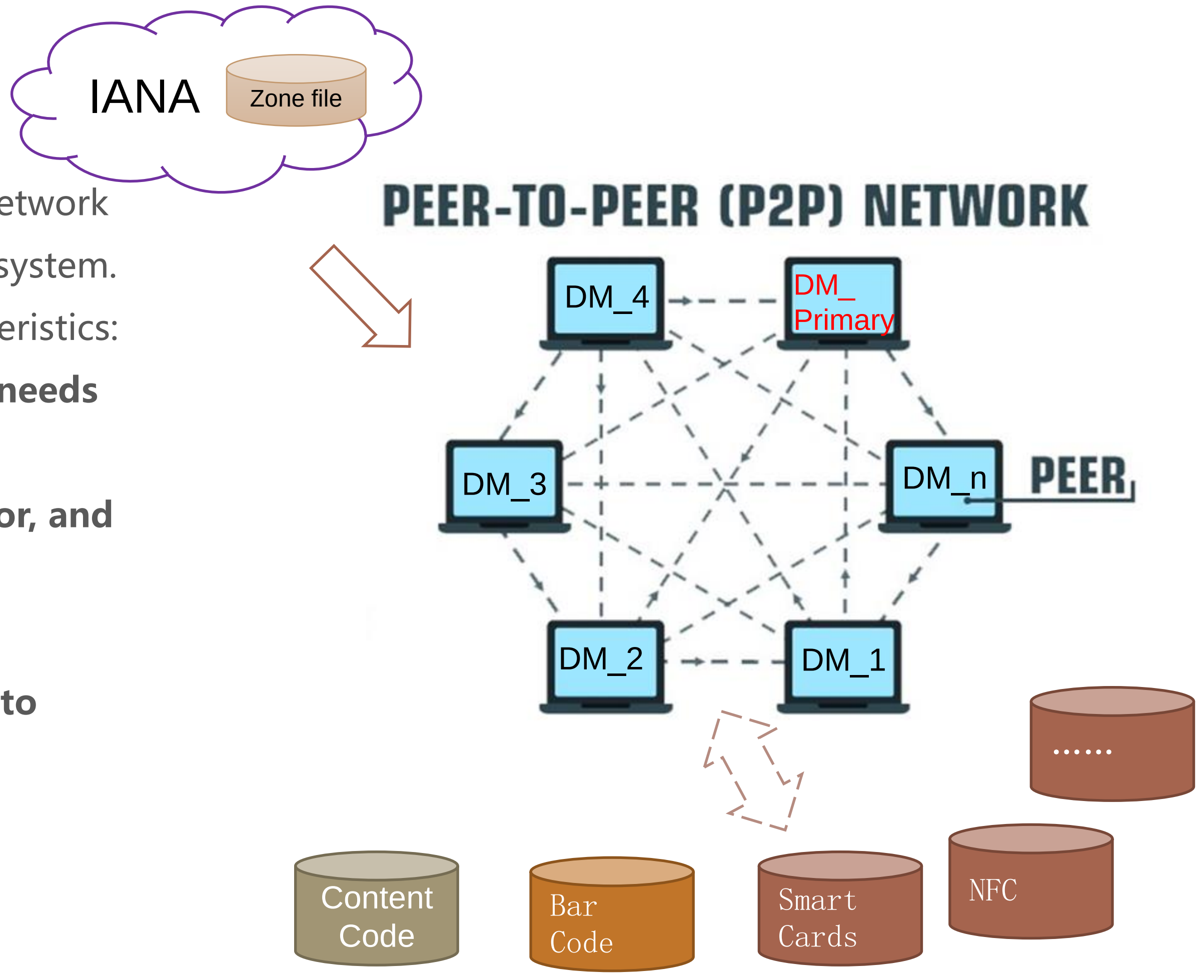
Yeti Phase-1

1.2 Project Overview

The Yeti Project Phase-2 is based on a P2P network and designs a new decentralized distributed DM system.

The system design has the following characteristics:

1. **Decentralized, no central node, each node needs to reach a consensus when performing operations, the Primary node is the executor, and has no special authority**
2. **Scalable，Increase system redundancy**
3. **Using threshold signature (TS) technology to reduce the number of DNSKEY**
4. **Introduced DM Management Committee (DMMC), responsible for transaction management**

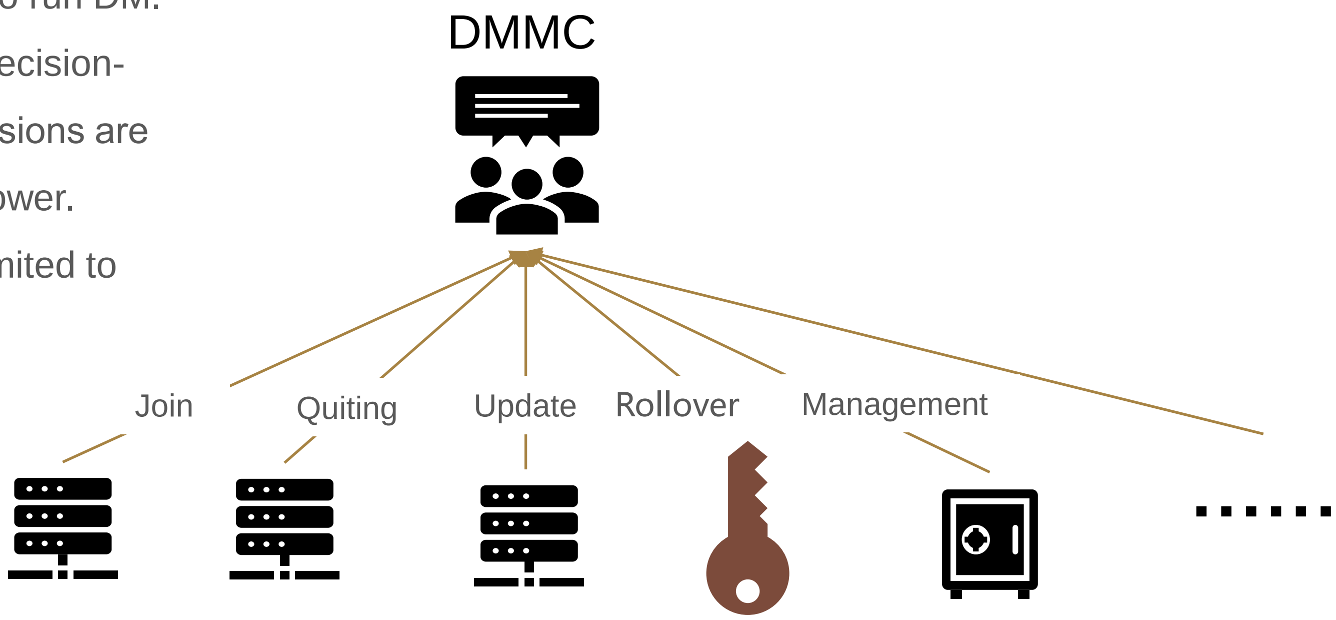


Yeti Phase-2

1.3 DMMC Introduction

DMMC is an organization that manages DM function and is composed of organizations who run DM. In the committee, all members participate in decision-making and planning. All the committee’s decisions are discussed collectively to avoid sovereign of power. DMMC's responsibilities include but are not limited to the following:

1. Approve the joining of new nodes
2. Approve the exit of the node
3. Approve the update of node information
4. Formulate KSK's rotation cycle
5. HSM management



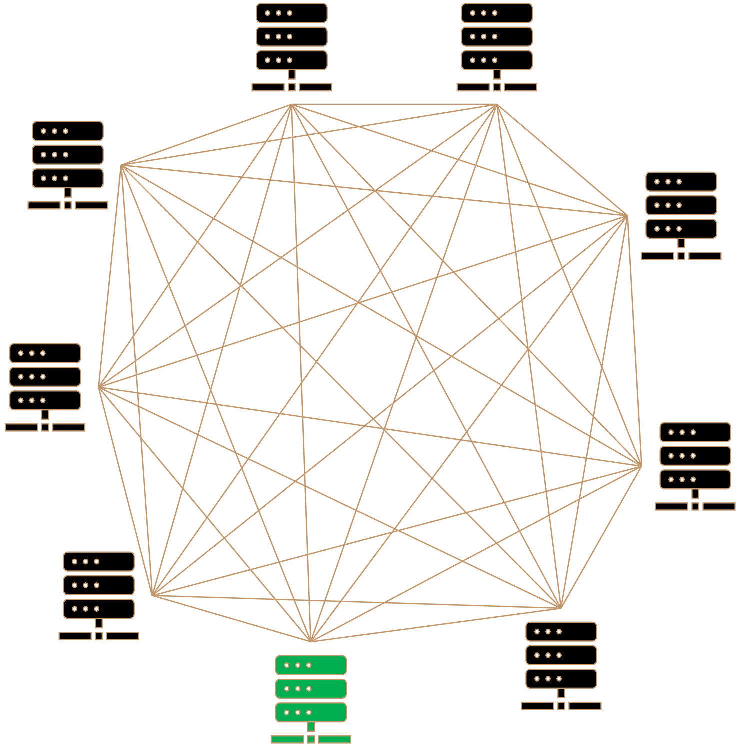
1.4 Network Topology and Algorithm

Yeti Project phase-2 system design draws on semi-distributed P2P network topology and blockchain consortium technology principles. The proposed Raft-like election algorithm and threshold signature algorithm are used to implement a decentralized multi-party trust distributed DM solution.

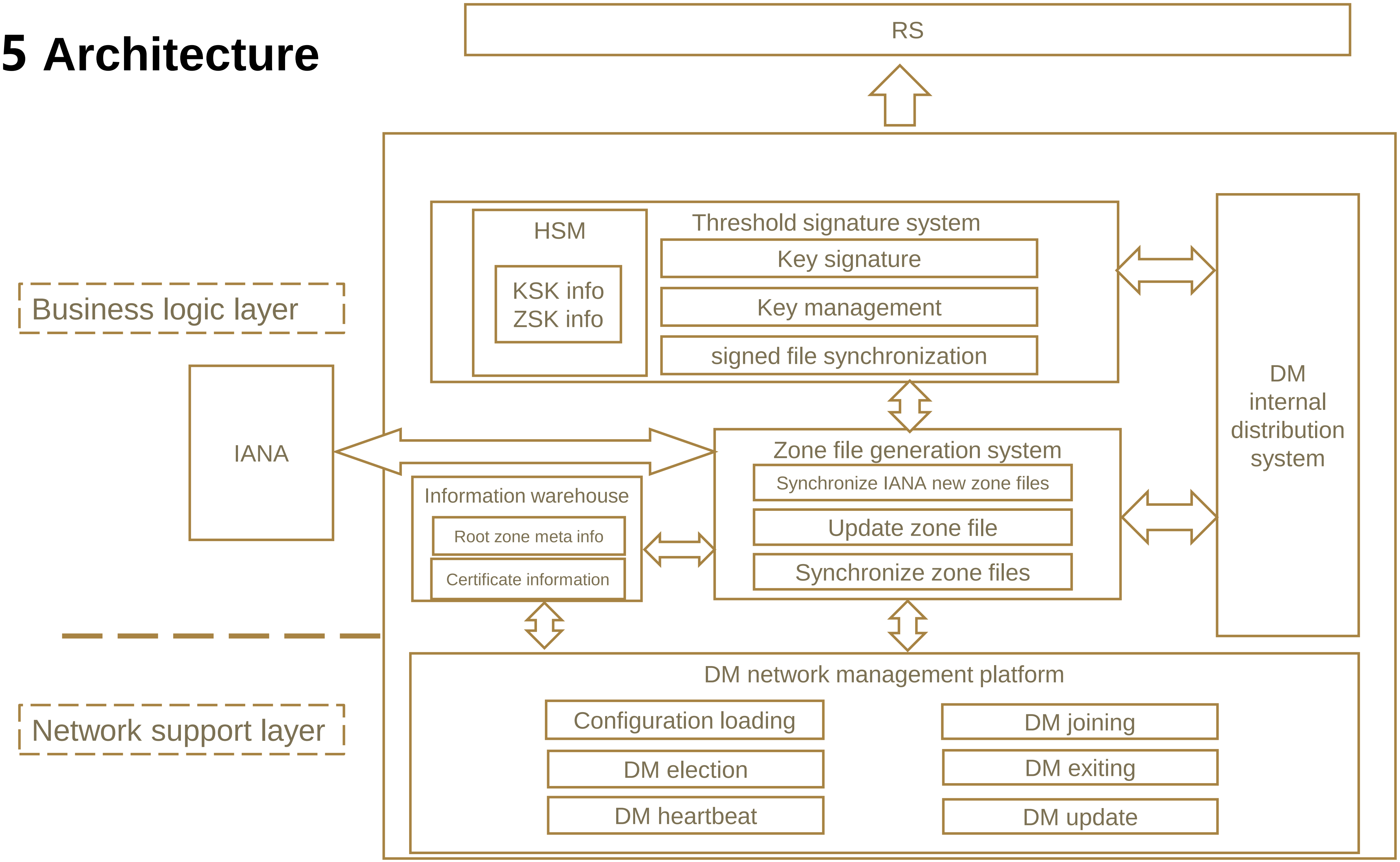
The semi-distributed network topology employs the advantages of a centralized P2P topology and a fully distributed P2P unstructured topology. It has good performance and scalability, and can be easily managed, but it is highly dependent on the Primary node , and the Raft algorithm solves the problem. The primary node dependency problem ensures that in the event of a failure of the Primary node, a new Primary node is elected to make the system function normally.

The system uses the admission mechanism and consensus mechanism of the blockchain consortium chain. The consortium chain refers to the blockchain whose consensus process is controlled by pre-selected nodes. Access is determined by the institutions in the alliance chain. The resolution is determined by collective voting and the practical Byzantine algorithm determines whether the resolution works.

Threshold signature effectively reduces the number of DNSKEY.

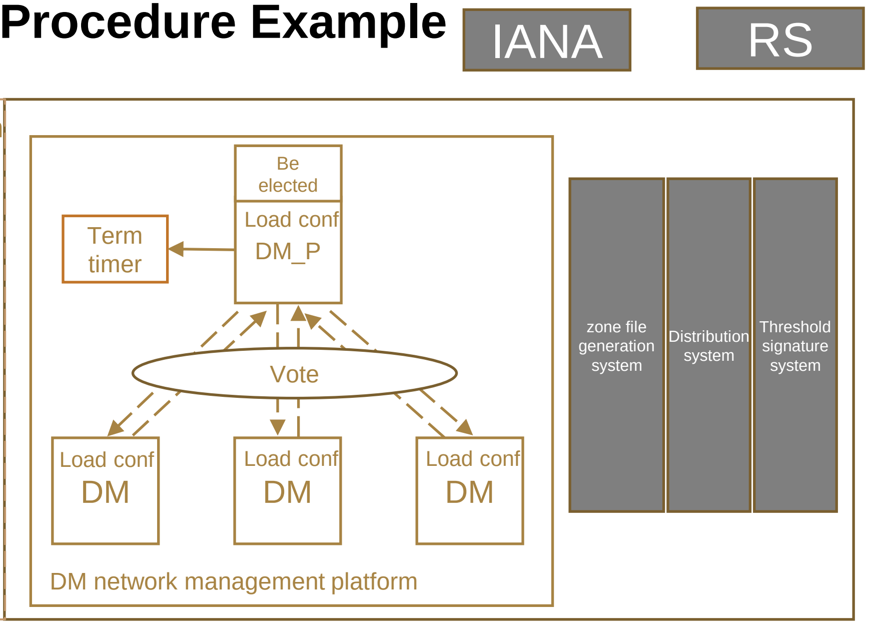


1.5 Architecture

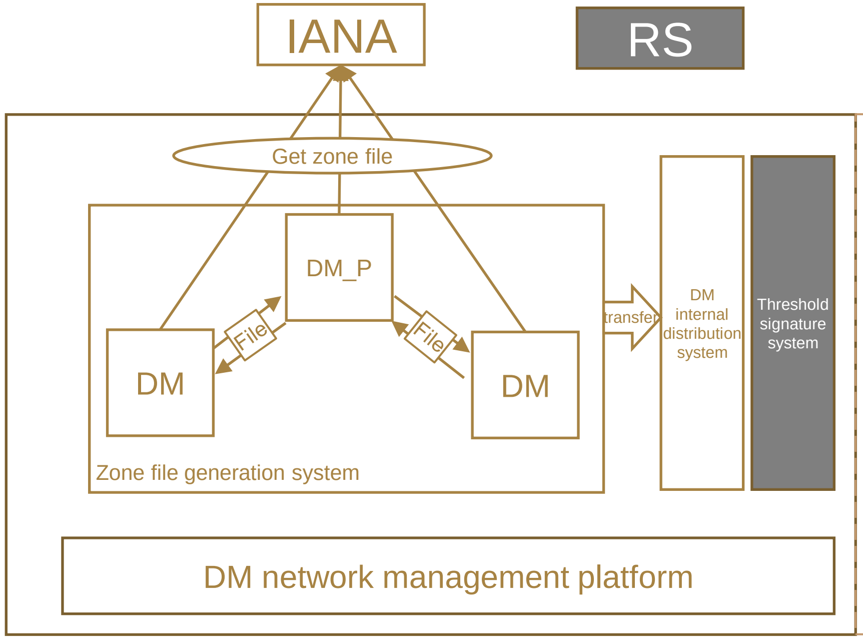


1.6 Running Procedure Example

Load configuration and certificate, elect Primary node, start tenure timer, and communicate with each other



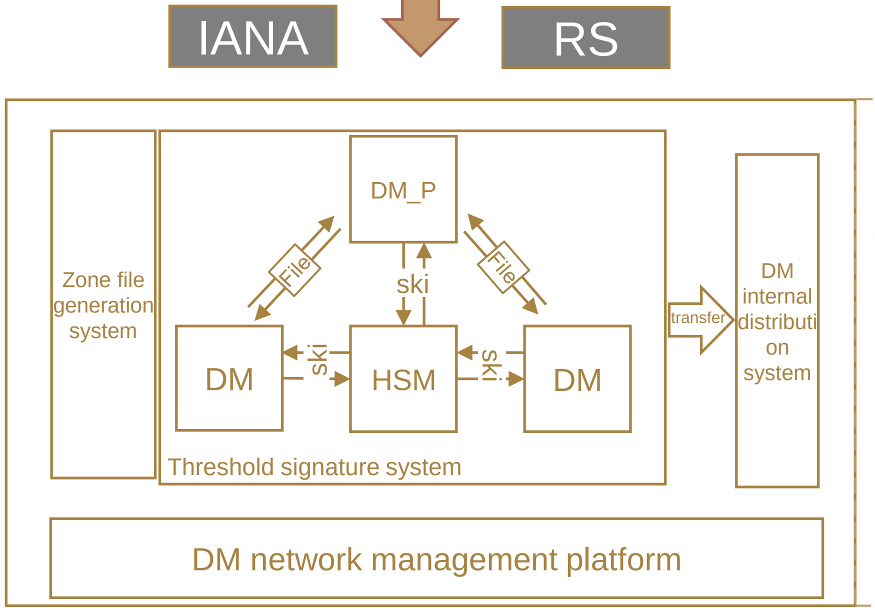
Startup state



IANA
RS
Get zone file
DM_P
File
File
DM
DM
Zone file generation system
transfer
DM internal distribution system
Threshold signature system
DM network management platform

Take the zone file, the Primary node is responsible for verification and distribution, to ensure that the zone file of each node is consistent

Get file status

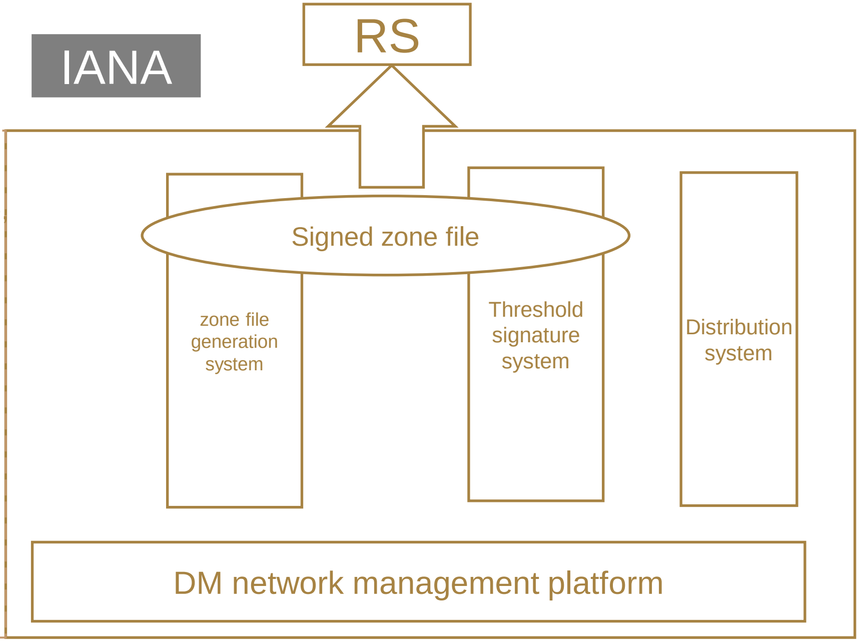


IANA
RS
Zone file generation system
DM_P
File
File
ski
DM
ski
HSM
ski
DM
Threshold signature system
transfer
DM internal distributi on system
DM network management platform

DM system performs threshold signature and synchronizes signature files

Signature status

After each node forms a unified file, it synchronizes with the root server. End one cycle and wait for the next cycle to start



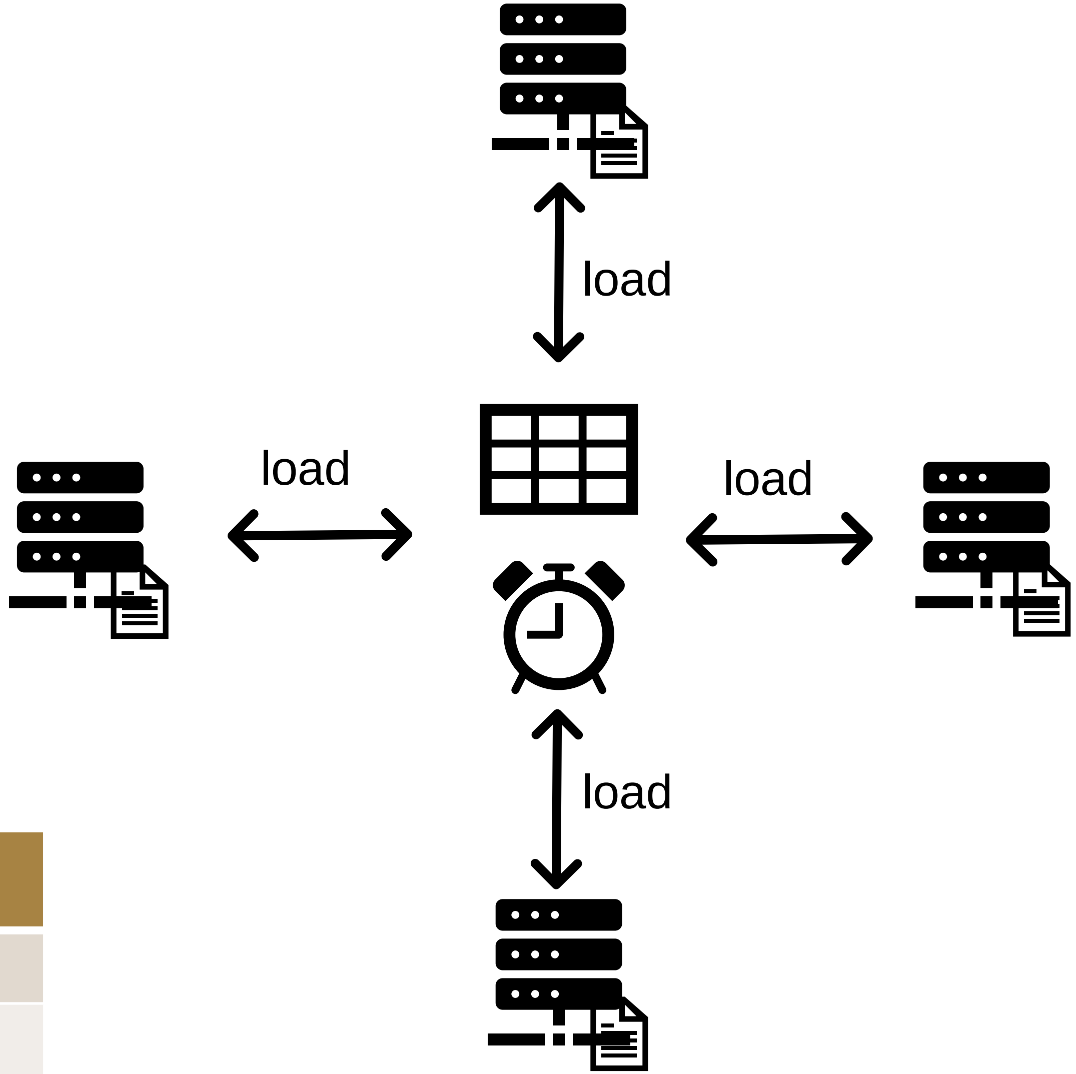
IANA
RS
Signed zone file
zone file generation system
Threshold signature system
Distribution system
DM network management platform

Root server synchronization status

/02 DM Network

2.1 Network Initialization

1. DM loads the initial configuration file, which is a persistent list
2. File timer initialization



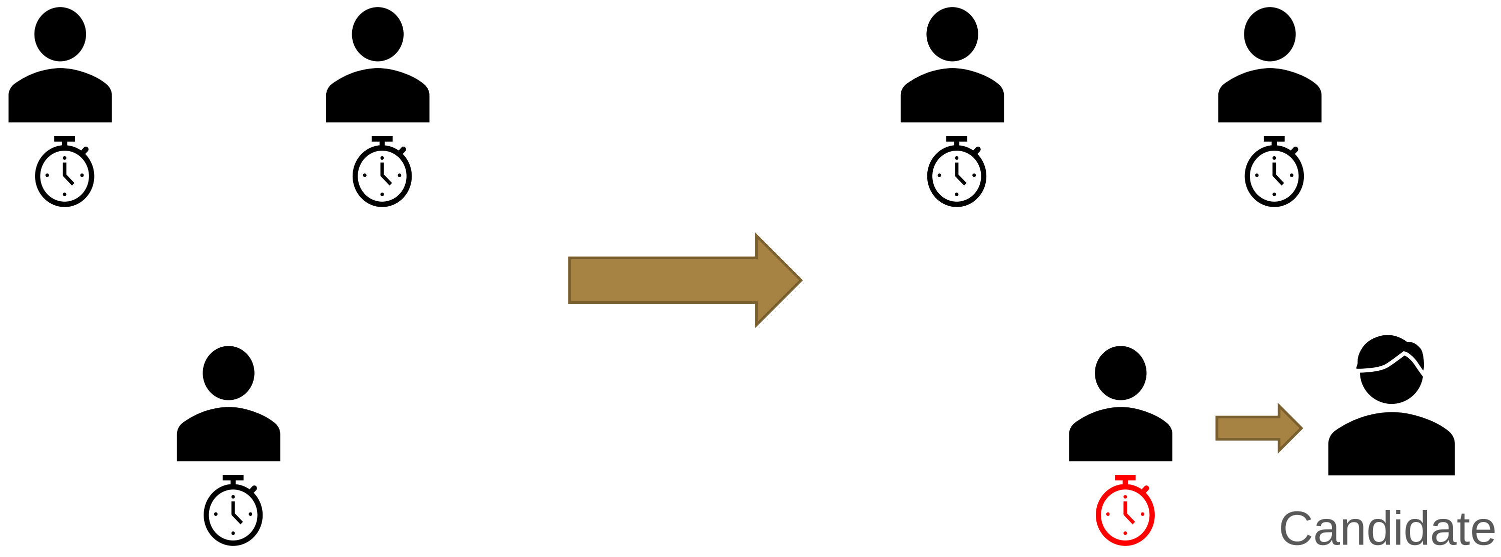
The configuration file format is as follows：

name	address	point	option
A	1111::1111	100	
B	2222::2222	100	

2.2 Election Part 1

Primary election

The system intends to use Raft's election and heartbeat mechanism. A new election rollover mechanism is added. The term of office is set to a fixed value. If the term is exceeded, the election is re-elected.

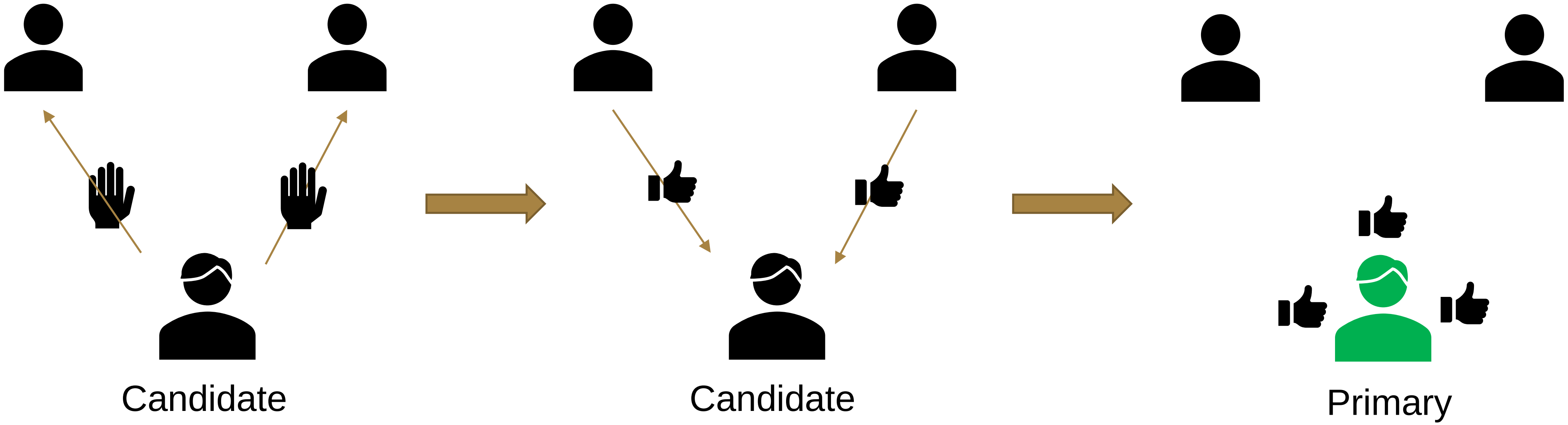


1. Each DM starts a random election timeout timer, time is random

2. The first node to trigger the timer becomes a candidate

Reference：
http://thesecretlivesofdata.com/raft/#overview

2.2 Election Part 2



3. Candidate vote for himself and initiate votes against other candidates

4. Follower vote according to FCFS principles

5. More than half of the votes, the election was successful

2.3 Reward and Punishment

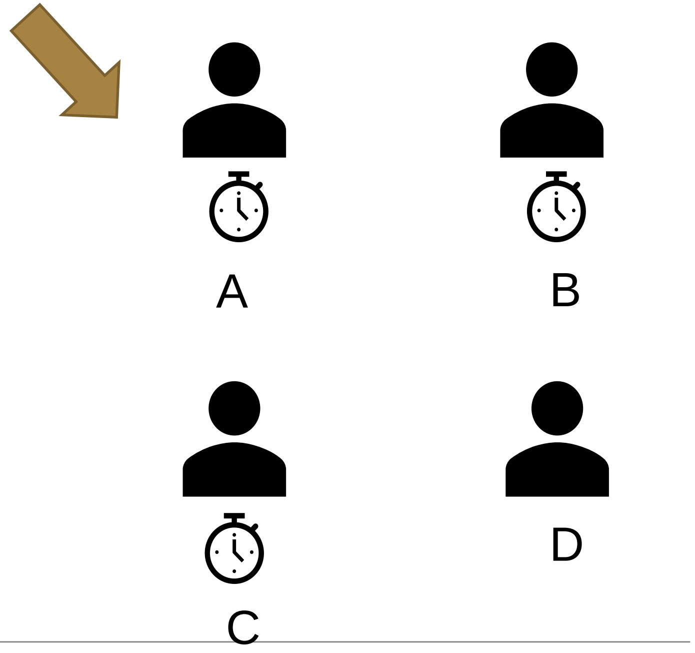
Reward and punishment mechanism

The primary node initiates sorting, and sorting depends on the principle of proof-of-work-efficiency. When node has high efficiency, then implements a plus point strategy, and if it has low efficiency, then implements a minus point strategy.

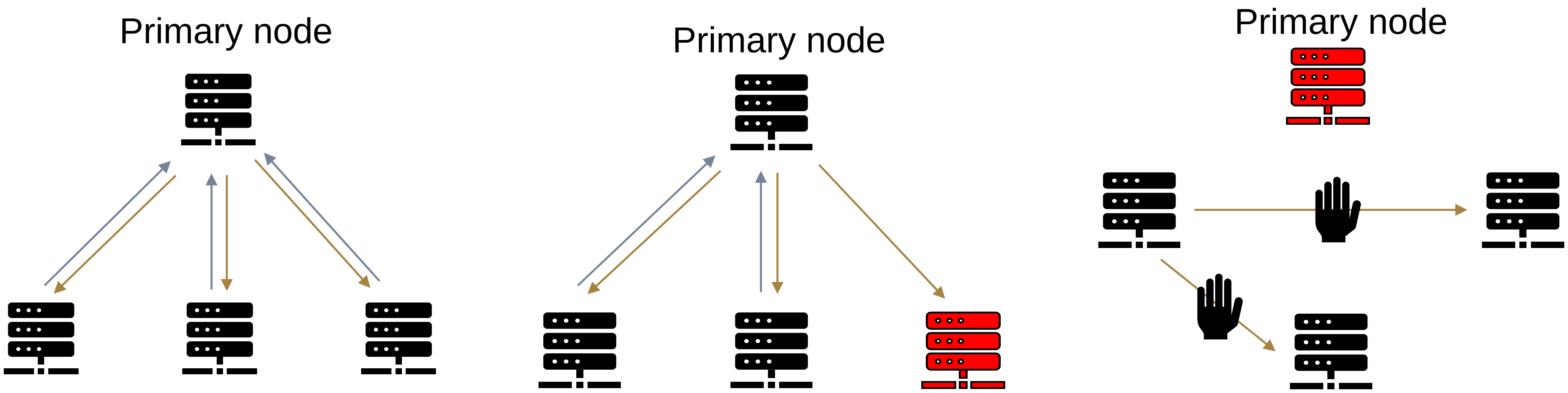
name	address	point	option
A	1111::1111	100	
B	2222::2222	95	
C	3333::3333	95	
D	4444::4444	50	

Election principle

The candidate node depend on the reward and punishment mechanism, and the point value serves as the basic basis of the node ordering principle. The Primary node is selected in the subset with a high point value. The non-subset nodes no longer participate in the election of the master node, but still have the voting power. After the master selection is completed, the node point value is initialized.



2.4 Heartbeat



Normal

Non-Primary node failure

Primary node failure

The working state of each node is maintained by the heartbeat detection mechanism. If the non-Primary node fails, the heartbeat detection mechanism continues to operate, and the ordinary node does not respond. If the Primary node fails, one of its supporters continues to vote as a candidate to become a Primary node.

2.5 DM Joining Process

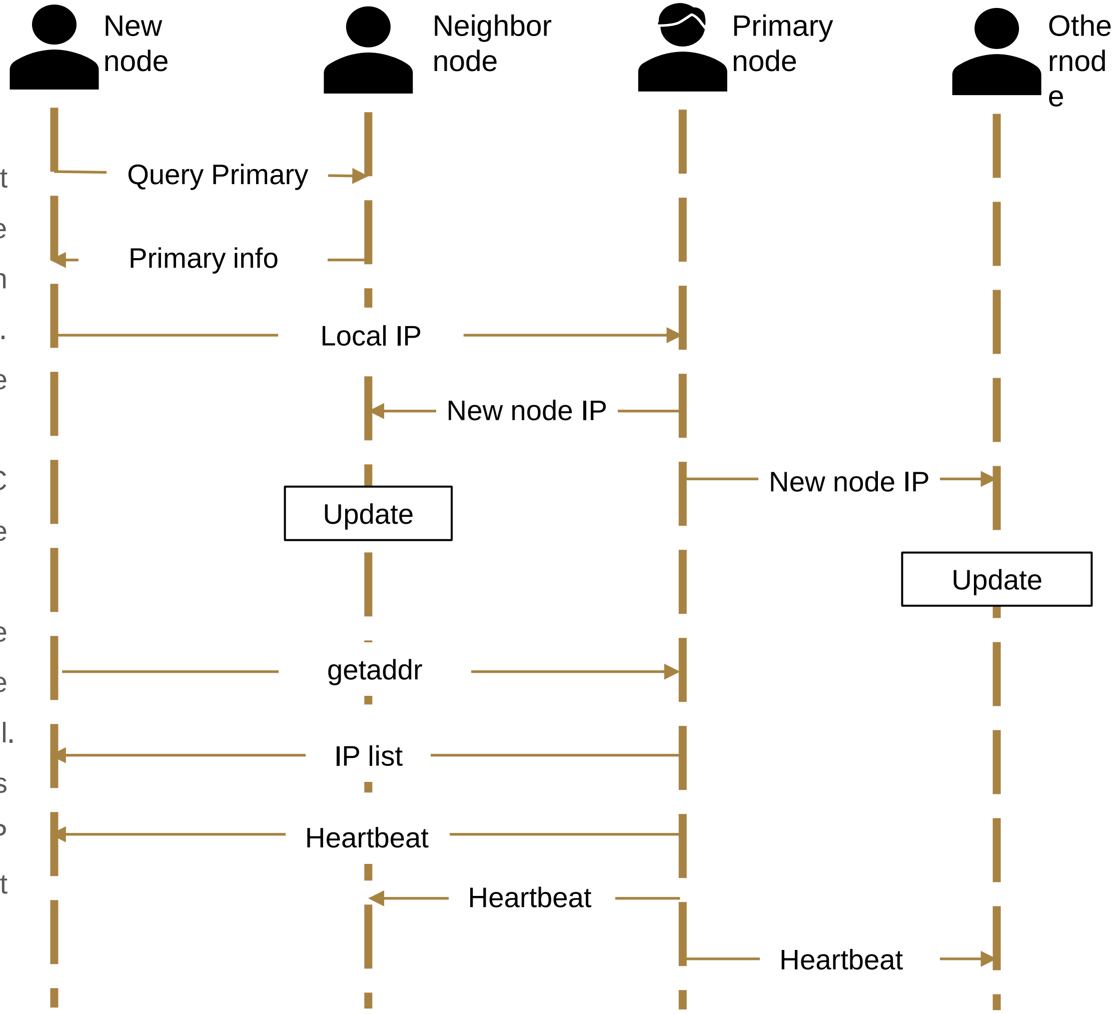
Node discovering

When a new node is started, in order to participate in collaborative work, it must have the function of discovering other nodes in the DM network. Since the DM network topology is not based on the geographic location between nodes, the geographic information of each node is completely irrelevant. When a new node is connected, the DM node existing in the network can be randomly selected to connect to it.

The DM network supports identity authentication. After the DMCC authentication approves, the corresponding digital certificate is issued to the new node, and the digital certificate is used to confirm the identity.

The new node itself has a list of IP addresses of the running DM. At the beginning of the connection, the new node randomly extracts some addresses as a subset and connects in turn until the connection is successful.

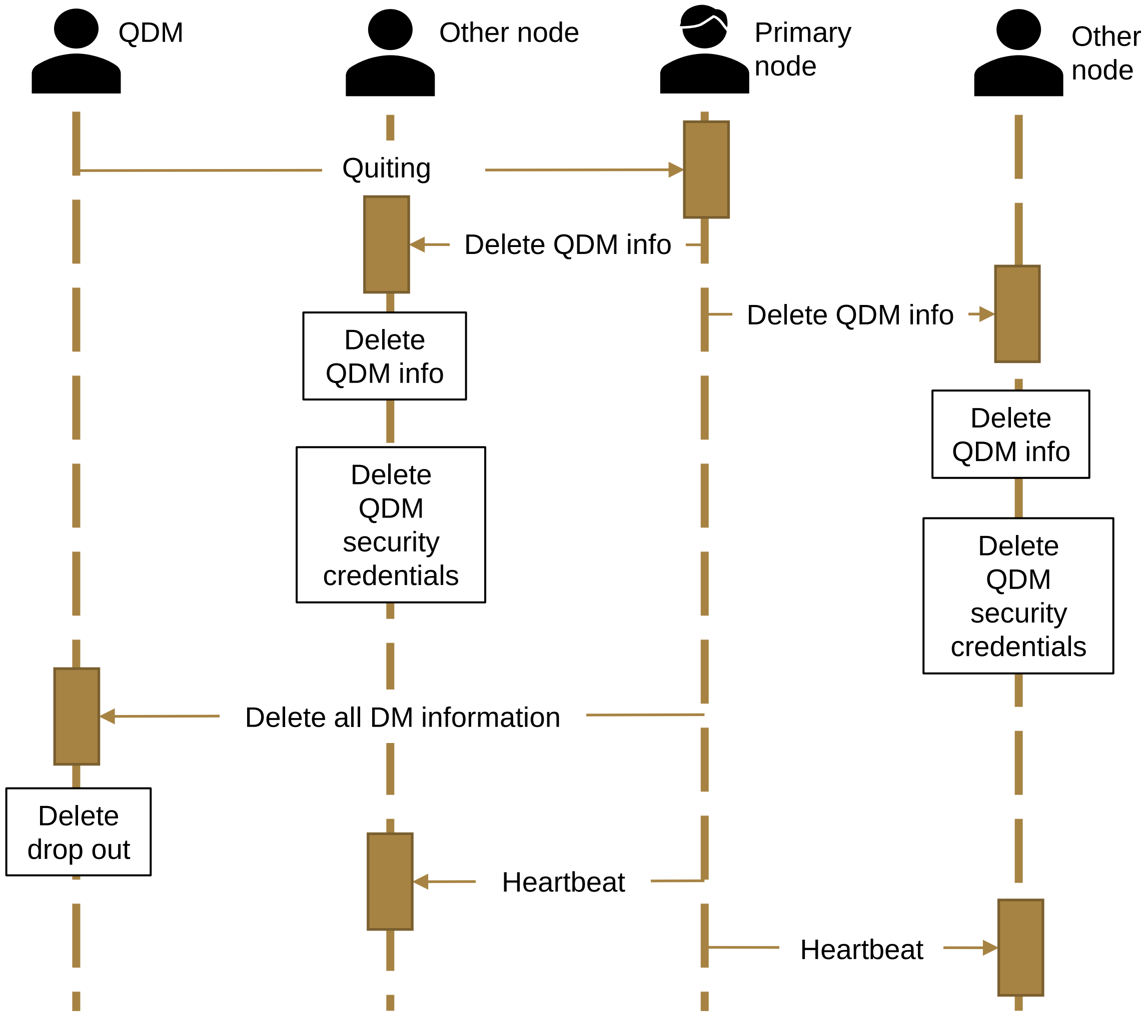
The new node queries the Primary node and publishes its own address information through the Primary node. The other nodes update the IP information list. The new node obtains the Primary node's IP information list to complete the synchronization. The specific steps are shown in the figure.



2.6 DM Withdraw Process

DM needs to have the function of exiting the DM network. QDM needs to delete all information related to itself in the DM network before withdrawing. Including addr information and security certificates retained in other nodes, its own IP address information list, and security certificates of other nodes.

According to the characteristics of the Raft algorithm, QDM knows the Primary node and sends an exit message to the Primary node. The Primary node distributes the deletion point information message to each node. After each node deletes the information, the Primary node sends a delete all node information message to QDM. The specific process is shown in the figure.



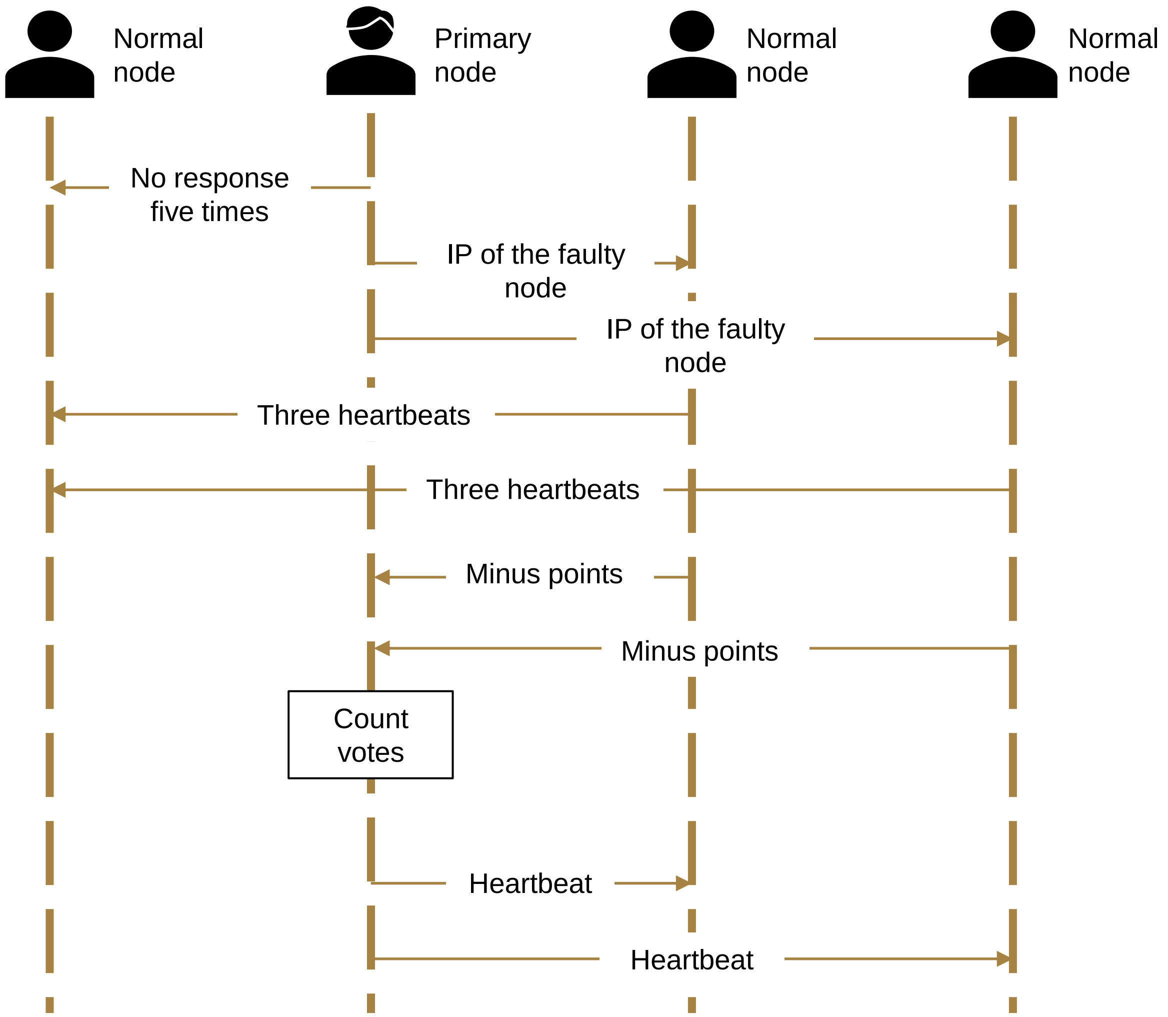
2.7 DM Kick-out Process

Automatic kick-out mechanism

The kick-out mechanism is based on the heartbeat detection mechanism. A node receives the detection of the Primary node. If the node does not respond, the point proportion is lowered. When it fails to response for five times, an automatic kick-out process is initiated.

The automatic kick-out mechanism is different from the DM's withdraw process. After the automatic kick-out, the node no longer participates in this round of transactions, but does not affect subsequent rounds. The kick-out of a node depends on the network conditions of the DM and its own performance.

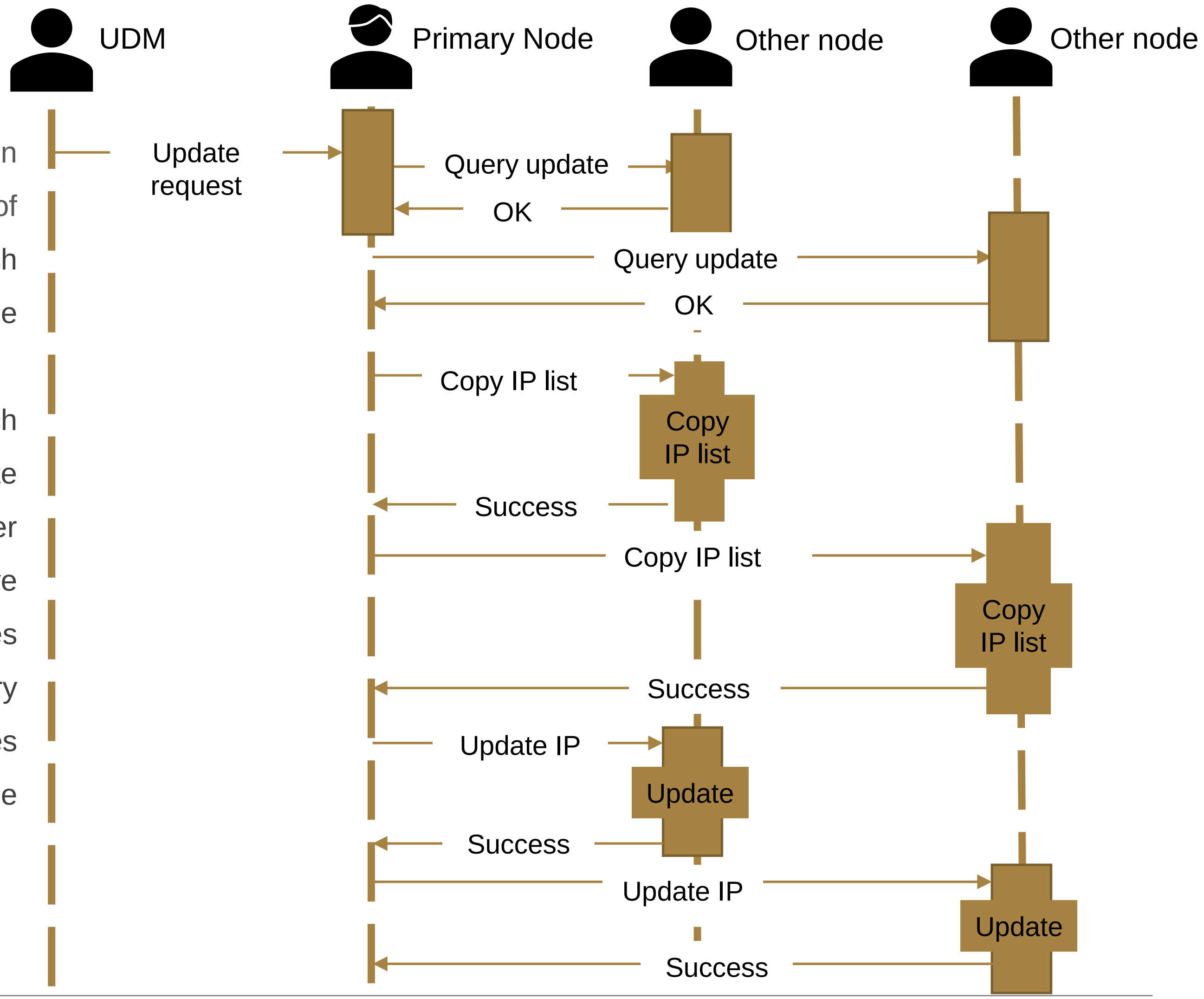
The Primary node informs each node of the address of failures, each node initiates a heartbeat to the failed node, records the response, gives a processing strategy, then the Primary node statistical results, and agrees to kick out more than 50% of the votes.



2.8 DM Update Process

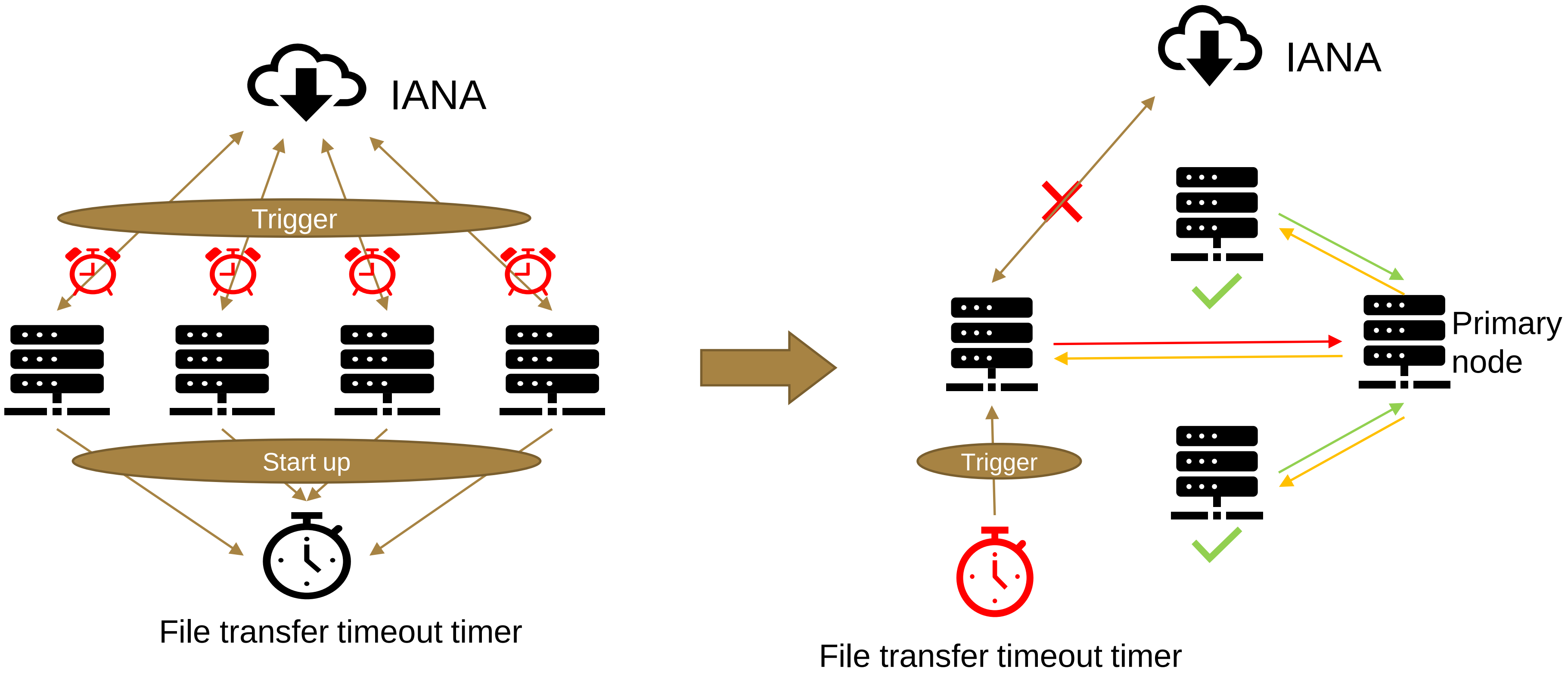
When DM nodes operate, the DM's own information may change. Therefore, it is required to be capable of updating. Due to the characteristics of the DM network, each node must ensure that the update happens only when the process can be conducted simultaneously.

The DM is updated using the 3PC protocol, and each DM will generate a copy of the old version after the update for backup. First, the Primary node asks each node whether it can update the transaction. After receiving a positive response, it enters the preparation stage. The node copies the IP information and waits for the update. The Primary receives the positive answer from each node and notifies each node to update. Among them, if any process fails, the transaction is rolled back.



/03 Zone File Generation

3.1 Acquisition of Zone File



To ensure that all DM zone file versions are identical and to obtain zone files at the same time, a file fetch timer is required; to ensure that the DM can disconnect from IANA and prevent the connection from taking too long, a file transfer timeout timer is required.

The file fetch timer starts after the system starts, the file transfer timer starts when the zone file transfer starts, and the Primary node receives the transfer completion message after the transfer is complete.

3.2 Zone File Generation Specification

Meta information processing

Each node processes meta information

- Eliminate IANA meta information

  DNSSEC (NSEC, RRSIG, DNSKEY)

  SOA, NS recording
- Keep top-level domain information
- Add Yeti meta information

  Yeti SOA recording

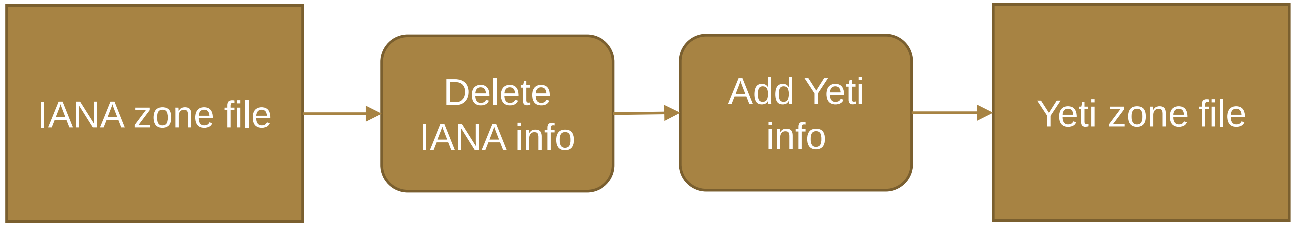
  Yeti NS、RRset recording

When adding Yeti meta information, under the premise of ensuring the same rules for generating zone files, there are two following schemes:

◆ Generate zone files for each node, then compare (recommended)

◆ Primary node generates zone files and distributes

After the meta information processing is completed, a complete zone file is generated



Note: The root zone meta information and security certificate are stored in each node to ensure that the file of each node is consistent, and the update uses a delay mechanism, which is the same as the 3DM scheme mechanism.

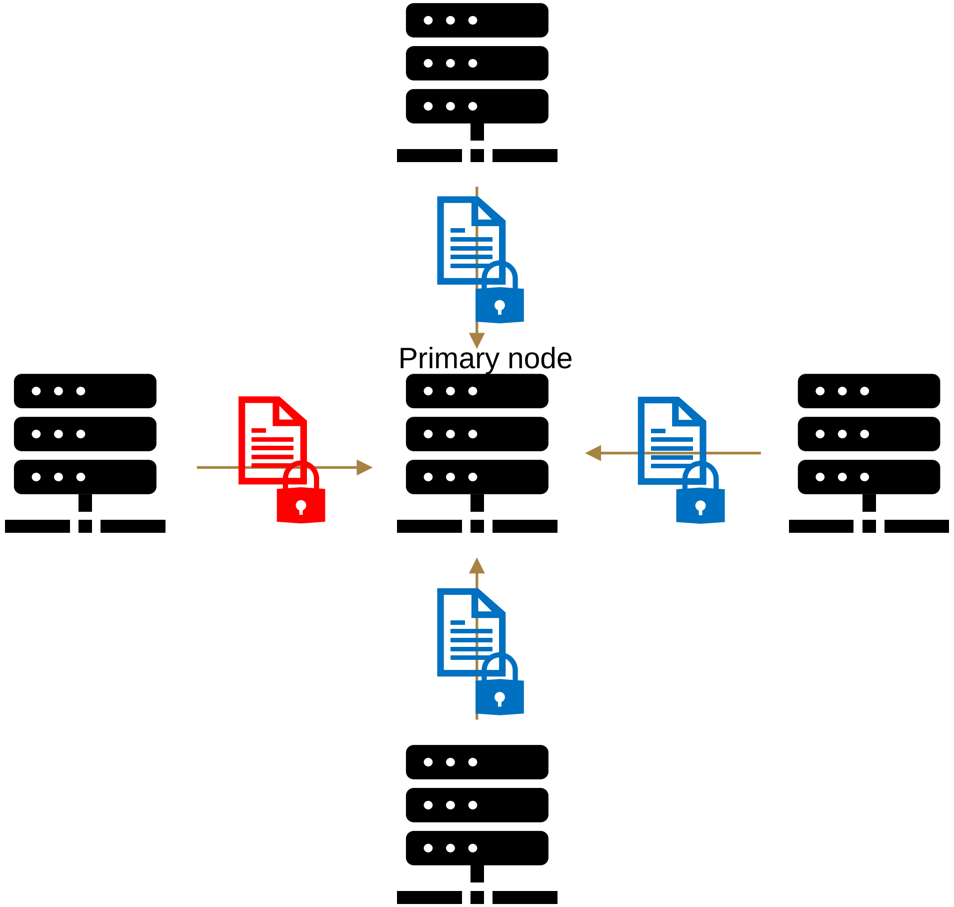
Detail： https://yeti-dns.org/alg-roll-test.html

Consensus reaching method is consistent with IP information list method

3.3 Consensus on Zone File

Verification

Each node adopts the hash algorithm to generate a summary for the zone file, define the message, send it to the Primary node, and compare the summary.



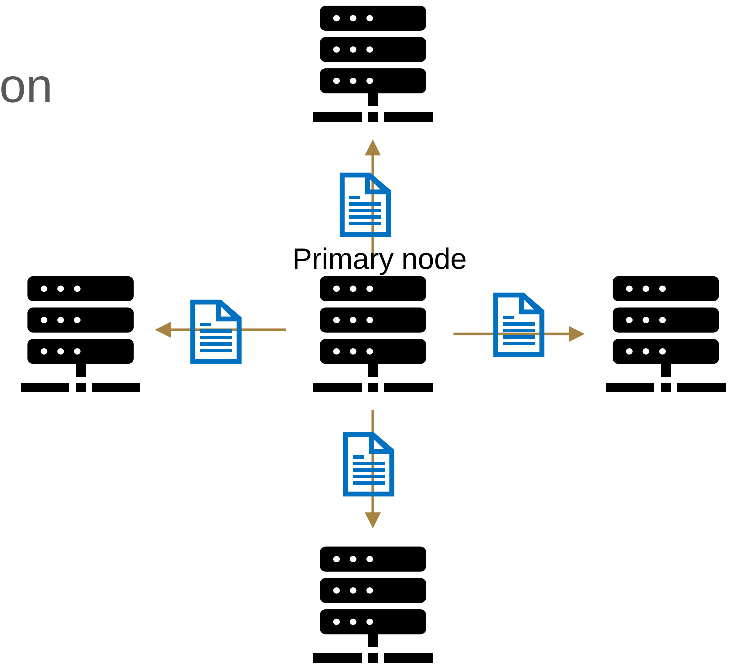
3.4 File Synchronization

Decision and synchronization

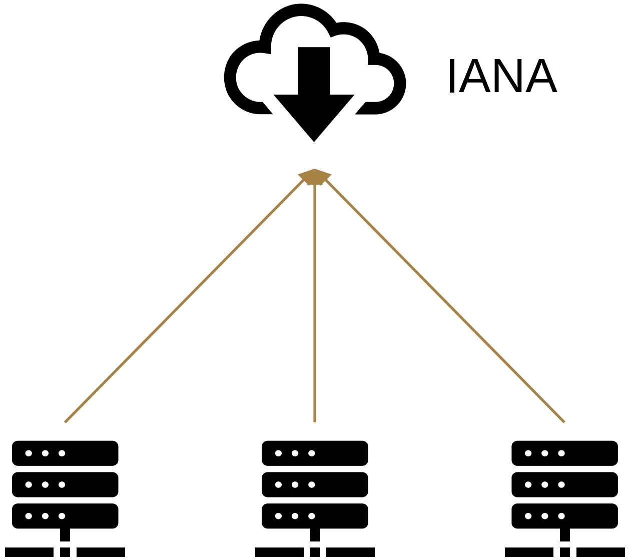
The decision-making mechanism draws on the practical Byzantine algorithm (PBFT). The core theory of the PBFT algorithm is n >= 3f + 1, n represents the number of nodes, f represents the number of inconsistent hash digests, and determines whether the expression is true:

True： Primary node distributes the correct zone file to each node

False： Primary node gives up operation and sends an acquisition message to all nodes



n >= 3f + 1



n < 3f + 1

/04 Threshold Signature

4.1 HSM

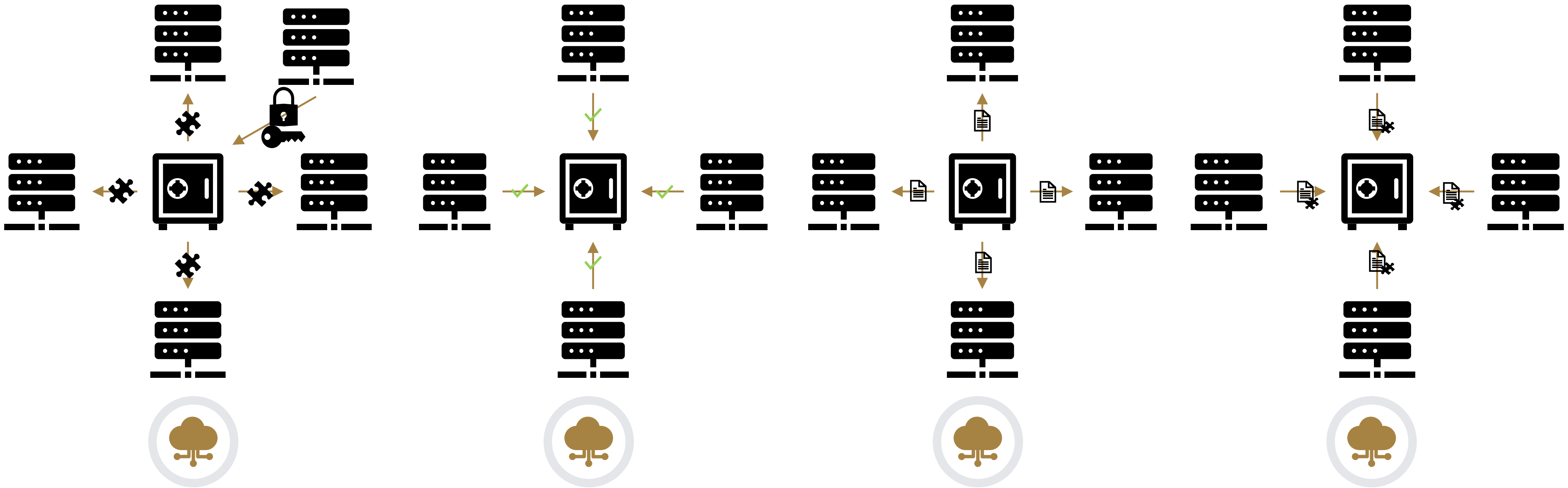
HSM is a system that provides a low-cost and highly secure solution for the key storage.

For sensitive information such as keys, HSM provides protection at the logical and physical levels to prevent unauthorized access and intrusion.

HSM provides tamper proof and tamper evidence functions. It is used in threshold signatures and is responsible for generating and storing keys used to sign DNS zone files. The management of HSM is carried out by DMMC, and two HSMs are set up to achieve high availability for the system.

- Security key generation
- Secure key storage and management
- Encrypt sensitive data
- Symmetric and asymmetric encryption calculations for offloading application servers
- ……

4.2 Threshold Signature



The Primary node uses the (t, n) threshold signature method to generate n private key shares ski through HSM and distribute to each node

Each node receives the private key share ski and replies with the confirmation message

参考：https://niclabs.cl/tchsm/

The Primary node uses the hash digest algorithm to generate a summary for the RR, and distributes the summary to each node

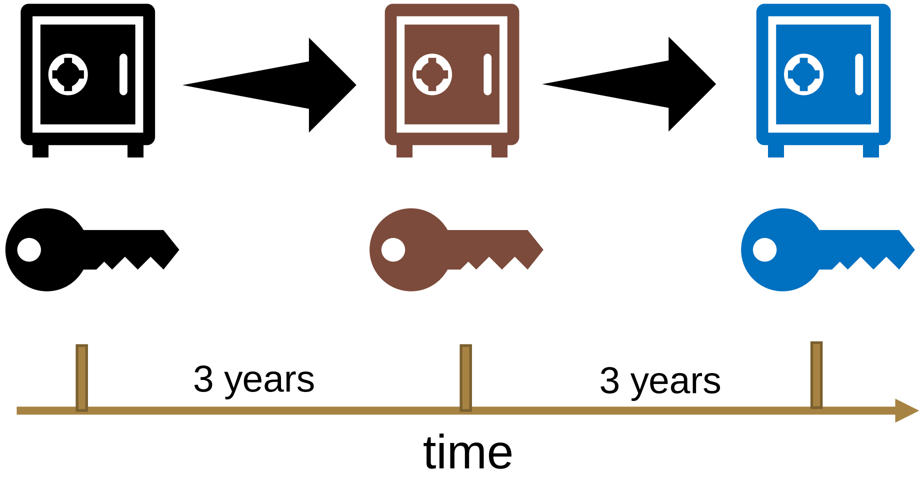
Each node uses the private key share signature, returns the signed summary to the HSM, forms a complete signature, and stores the signed file to the Primary node

4.3 Key Management

Key rollover

Threshold key generation, constructing a distributed key generation protocol (DKG) based on security parameters, the protocol runs to output a common public key pk and private key shares ski belonging to different parties, and gather together to meet the threshold number The private key share can construct a real private key sk, and the public key as an important proof of identity verification needs to ensure its security.

DNSSEC uses the short-term key ZSK to periodically calculate the signatures of DNS records, while using the long-term key KSK to calculate the signature on ZSK. Therefore, KSK is saved to HSM for a period of three years.



/05 Signed Root Zone File Synchronization

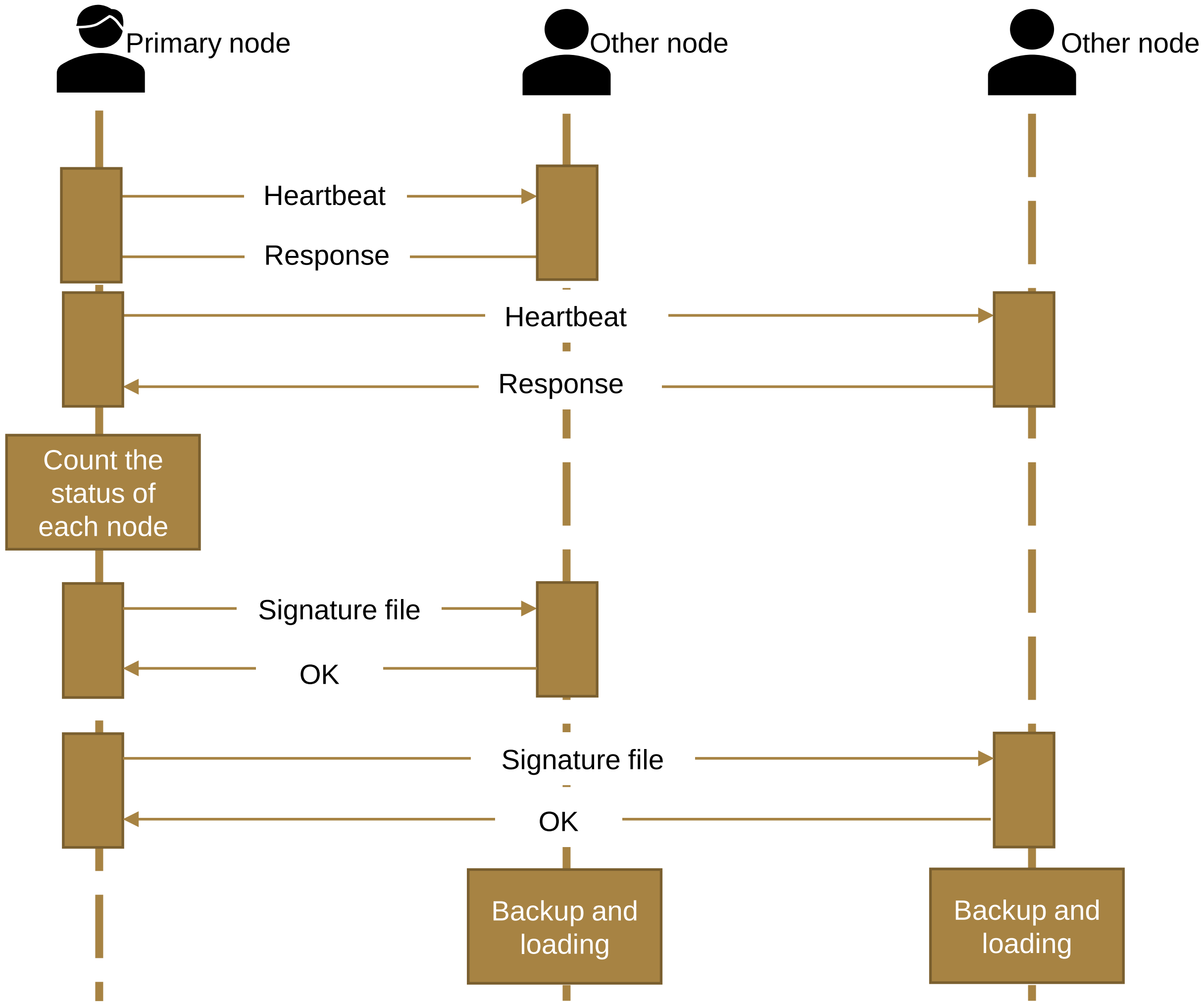
5.1 Signed File Synchronization

Synchronization

To prevent the zone file from bifurcation, after the Primary node generates the correct signature file, then it will be synchronized to each node to ensure that the version of each node is unified and effective.

1. Primary node initiates heartbeat detection mechanism
2. Regular node response
3. The Primary node determines the working status of each node and distributes signature files to all nodes
4. Each node completes the file transfer and replies with a confirmation message
5. Each node backs up the old version and loads the new version

Note: All nodes of the signed file will be distributed, regardless of whether the previous node correctly obtained the file



/06 DM and Root Server Synchronization

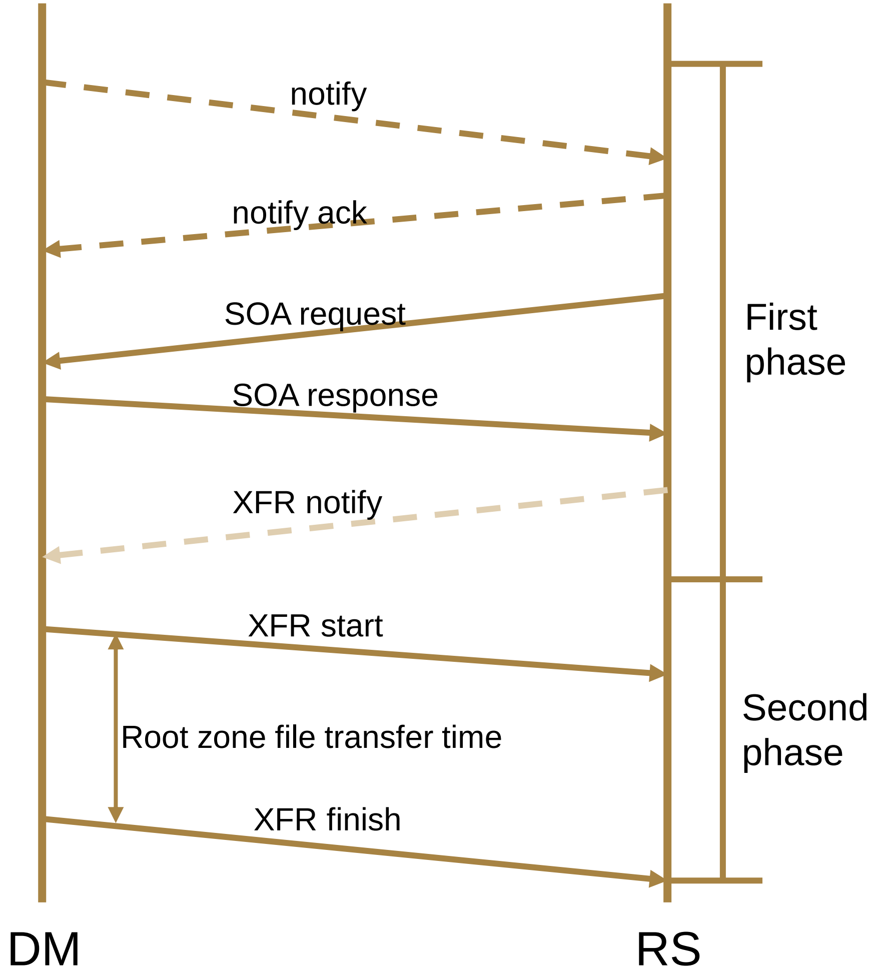
6.1 Root Server and DM Synchronization Scheme

Solution 1: Separate synchronization between root server and DM

The synchronization between the root server and the DM uses the original synchronization scheme, that is, the RS randomly sends an SOA request to some of the DMs configured by itself. The DM responds with the current root zone file serial number, and decides whether to perform zone transfer based on the serial number.

Solution 2: Root server and DM convergence synchronization (recommended)

DM and root server are merged to replace root server. The root server and the DM system are different processes. The zone transfer between the root server and the DM only occurs on the local machine, which greatly improves the transmission efficiency and success rate, and also reduces the synchronization time. However, as the DM expands, the root server expands accordingly, leading to an increase in packet length.



/07 DM Security Design

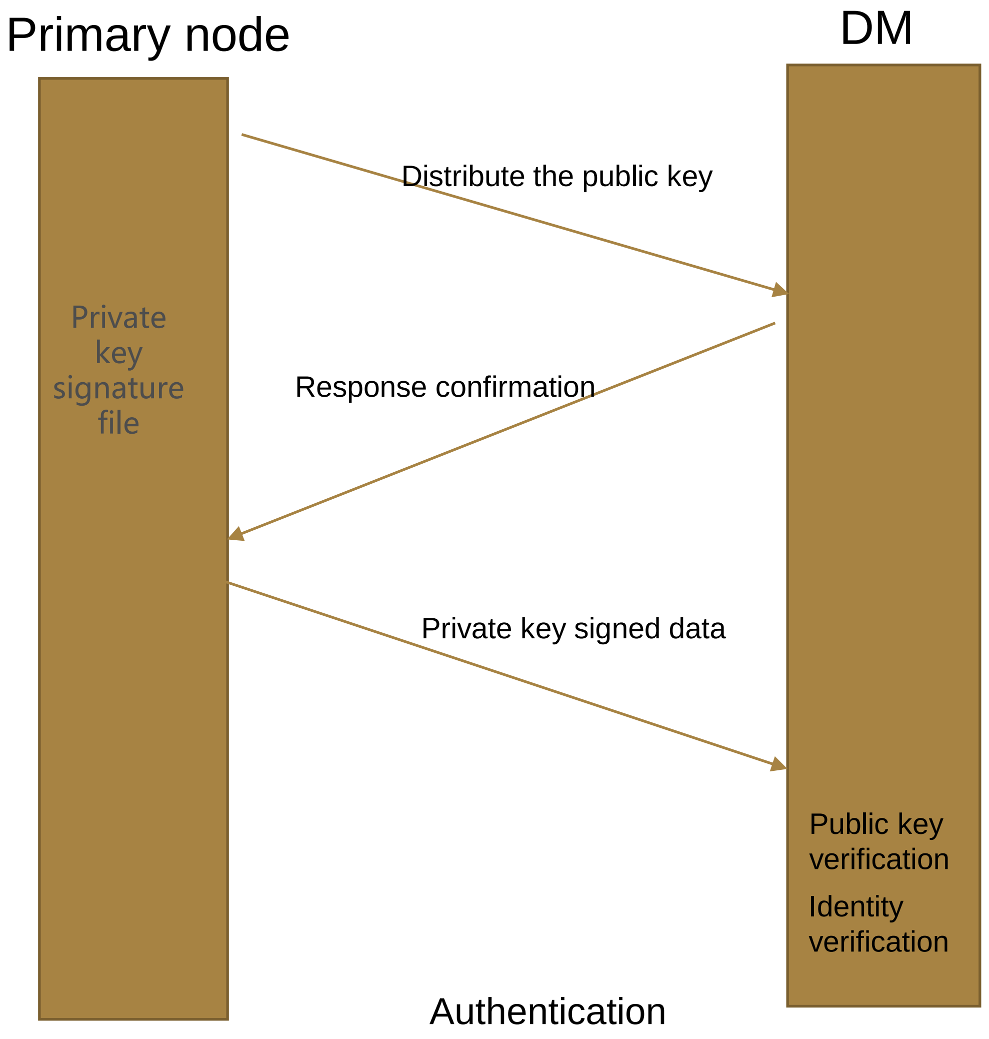
7.1 Security Considerations

Encryption

TLS certificates are applied between DMs to ensure communication security. When the network is initialized, each initial node contains a corresponding public key certificate, and when a new node is added, the corresponding DM certificate needs to be authorized to ensure that the nodes trust each other.

Authentication

Taking the Primary node as an example, the control message is signed with the Primary node's private key. Each DM has a public key corresponding to the Primary node. When the message signed by the Primary node's private key is sent to the DM, the DM uses the Primary node's public key for identity verification that the update messages have been sent are sent by the Primary node.



THANKS

2020.06.18

www.yeti-dns.org