Introduction

The concept for the system came from a real-world requirement. While the initial concept did not include using a distributed mode I quickly realised employing a distributed model would greatly enhance the concept introducing reliability that would not be possible in a single system structure.

The requirement for the system stems for some work I did for a small IT support company. The company focuses on the SME environment with most clients typically running one or two internal servers and the IT support company being the sole provider of IT services i.e. no internal IT staff in the companies themselves.

At one customer, purely by chance the IT support company noticed the backups had not run in months, while this issue was reported in the system logs no one was checking them. The nature of small businesses and the margins involved mean many of them do not pay for active/ preventative IT support but rather employ a break-fix type model so a failing backup such as the one mentioned could potentially go unnoticed for months or until a disaster strikes and a backup is required.

The concept of the system stemmed from this event, the idea is to provide a simple reporting system which could catch events like this, ideally the system can be implemented with very little cost to both the small business or the IT support company. The following section will describe the key requirements of the system in more detail including the distributed nature of key components.

Statement of requirements

Functional Requirements

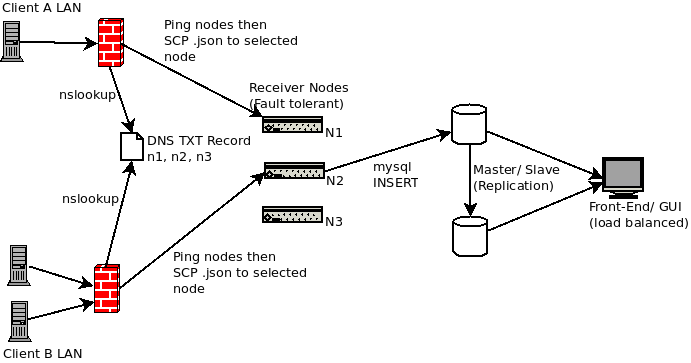
1. The system will log all backup events from customer Windows servers.
2. The system will allow viewing of these backup events without remote access to the customers site.
3. Receiving nodes can be added or removed from the system without requiring any configuration changes on the customers site.
4. Clients will cache logs and retry if no receiver nodes are available.

Non-functional Requirements

1. The system should be able to tolerate at least one receiver node failure, there should always be at least n + 1 nodes setup.
2. Customer setup needs to be simple with no software install if possible.
3. Client systems should log at least one a day even if there are no backup logs.

System Requirements

The system as a whole can be broken down into three key components. The client, the receiver nodes and the database. The diagram below shows how the components work together and the proceeding section explains the components in more detail.



INSERT CORE REQUIREMENTS OF THE SYSTEM – high availability (u**se some distributed concepts**)

The Client

The Client in the system will be the running on any servers within the IT support companies client offices that need to be monitored. These will be Windows Server OS machines exclusively. Some key aspects of this component are the following.

Must be easy to install, there needs to be very little configuration required for this as the IT Company needs to be able to offer this set-up at little or no cost to the client.

The Client component needs to be set-and-forget as the nature of the IT Support companies interactions with the company mean they would no be on site very often. For that reason we will use built in Operating System features where possible.

The communication from The Client to the Receiver Nodes needs to be outgoing from The Client network only. The network structure of these clients is generally very basic with no DMZ and very simple configuration on Firewalls. To avoid introducing security risk by opening ports in the firewall, all communication will be initiated by The Client.

Receiver Nodes

The receiver nodes sit outside all client sites. They can run on a private LAN or on a hosted VM. There are a few key requirements for the receiver nodes in order to improve availability of the overall system.

-There should be a minimum of 2 receiver nodes, ideally more.

-Receiver nodes should be hosted on different hardware, ideally in different locations, better still with different providers.

-Receiver nodes should only accept traffic from approved IPs (client site IPs)

-Setup of a receiver node should be a simple process, ideally running a single Bash script