1. ScopeF

This specification describes the features and specifications of the Feeder Automation Manager self-healing software.

2. ArchitectureF

The self-healing system must have the following broad capabilities:

2.1. The self-healing software shall support a substation decentralized communications architecture as well as a server based centralized communications architecture capable of automating thousands of switches per server deployment

2.2. The self-healing system shall support full data concentrator functionality capable of integrating thousands of connections and at least 500,000 data points per deployment

2.3. The self-healing system shall support a user configurable interconnectivity model input with graphical user interface

2.3.1. The user shall not need to script the system for individual events that might occur and should instead be able to configure the system so that the real world power system is represented in the configuration to the self-healing system

2.4. The self-healing system shall have a user configurable data collection system with graphical user interface

2.5. The self-healing system shall have a user configurable system to visualize the electrical system

2.6. The self-healing system shall have diagnostics tools for analyzing data and events

2.6.1. Real-time data views

2.6.2. Historical data views

2.6.3. Post event reports

2.6.4. System reports that include all logging and configuration parameters integrated together in one transportable file for streamlined remote support

2.6.5. Real time communication protocol analyzer

2.6.6. System event notification mechanisms

2.6.7. Post event diagnostics summary

2.6.8. Real-time operator status feedback of event activities

2.7. Logging mechanisms and log visualization tool

2.8. Simulation system for determining correct configurations

2.8.1. Simulation system should interact with the actual automation system logic and not just a representation of it

2.8.2. The automation system logic should not be aware if it is running in simulation or production to ensure the simulation is a true test of the logic and configurations

2.8.3. System should be capable of running in a mixed mode status where the system signals, such as current and voltage are simulated, but the device status feedback is provided via real control devices for laboratory and training setups

2.8.4. System should be capable of running hardware–in-loop testing for complete end to end tests

2.9. The system shall be capable of being managed at several levels of the system. These levels include:

2.9.1. The global-region level

2.9.1.1. Changes made at this level shall be the default

2.9.2. The subsystem level

2.9.2.1. Settings at this level shall be inherited from the global site level unless specifically defined at this level

2.9.3. The feeder level

2.9.3.1. Settings at this level shall be inherited from the subsystem level unless specifically defined at this level

2.9.4. The device level

2.9.4.1. Settings at this level shall be inherited from the feeder level unless specifically defined at this level

2.10. The self-healing system shall be capable of interacting with Distributed Energy Resources (DER) that may be present on the distribution system

2.10.1. The system shall be capable of accounting for bi-directional current flows associated with DER

2.11. A zone shall not have a limit of the number of devices that can bound its topographical area.

2.12. The topology of the system shall not need to be in a fixed or predefined state for the automation system to be active and functionally ready to respond to events.

2.12.1. Normally Closed and Normally Open devices are defined only so that the system can perform a Return to Normal (R2N) command upon request

2.13. The self-healing system shall be able to handle multiple faults in the same area

2.13.1. As long as there are viable sources still available that have sufficient capacity, the system shall be capable of performing restorations to utilize that capacity

2.13.2. The system shall be capable of automatically disabling itself at different levels if this multiple fault handling capability is not desired

2.14. The self-healing system must be able to support a standard set of integrated PLC type functions, including but not limited to: Basic Boolean functions such as Inverter, logical AND, OR; Best OF functionality for choosing redundant data points with best quality, Input Latching, Binary Debouncing, and Grouped Control in order to accommodate any special request functionality that is not part of the standard automation functions. These functions must be able to be executed on the standard data sets from the devices

2.15. The self-healing system shall support multiple communications media including serial and Ethernet radios, fiber, cellular modems, and others

2.16. At a minimum the communication portion of the self-healing system must be capable of supporting the following protocols:

- DNP3 Master
- Secured Authentication V5 for DNP3 Master
- SMP Interconnection
- ICCP Master
- DNP3 Slave
- Secure Authentication V5 for DNP3 Slave
- SMP Interconnection
- ICCP Slave

3. Field Device Interoperability

The self-healing system must be capable of interoperating with many different controls directly. These requirements ensure that the self-healing system is capable of leveraging the existing field devices deployed now and in the future, without requiring them having to be vendor specific.

3.1. The self-healing system shall support direct communication with field IEDs

3.2. The self-healing system shall support in-direct communication with field IEDs, via Remote Terminal Units (RTUs), Front End Processors (FEP), Gateway communication devices, SCADA, or other systems. These systems should not be required, but the self-healing system shall be able to gather data from these systems if available

3.3. No intermediate devices or logic processors shall be required for interoperability

3.4. No vendor specific hardware shall be required for field devices

3.4.1. Self-healing software must NOT lock the utility into a vendor specific hardware solution.

3.5. Forward and backward capability with field devices

3.5.1. No change out of field devices required for system upgrades

3.5.2. No firmware upgrades of field devices required for system upgrades

3.6. The self-healing system must be able to work with different data sets from the same device types

3.6.1. Example would be a device that has, no voltage, source side only voltage, low side voltage only, or voltage on both source and load sides.

3.7. The self-healing system must be able to interact with Electro-mechanical relays

3.8. Devices of differing fault and load interrupting capability shall be able to be incorporated into the system

3.8.1. Devices that can interrupt fault and load shall be incorporated

3.8.2. Devices that can detect faults but can only interrupt loads shall be incorporated

3.8.3. Devices that cannot detect or interrupt, loads or fault, shall be incorporated

3.9. Device data shall be able to be incorporated from non-switching devices to add in restoration decision making

3.9.1. Substation transformer loading shall be incorporated to ensure that a transformer will not be overloaded when accepting additional load

3.10. Systems shall be established by the self-healing vendor to ensure interoperability with existing field devices and to be able to incorporate new field devices into the existing system upon request

3.10.1. New field device integration on demand

3.11. The following is the minimal set of devices that the self-healing software shall support:

ABB PCD	S&C 5801
Cooper Form 4C	S&C 5802
Cooper Form 4D	S&C 5803
Cooper Form 5	S&C 6801
Cooper Form 5 Triple-Single	S&C 6801M
Cooper Form 6	S&C 6802
Cooper Form 6 Triple-Single	S&C 6803
Cooper Form 7 Triple-Single	S&C Intellirupter
Cooper iDC (DAS Controller)	SEL 351R
Cooper iDP-210 Relay	SEL 351S
Cooper iMC	SEL 451
Electromechanical Relays	SEL 451 (dual breaker)
GE F-60 Feeder Relay	SEL 651R
GE DART	SEL 751A
Generic Devices	Telvent RTU Switch Control
Noja RC10	Reliatronics R2000 Inertia Switch Control
Nulec CAPM	

4. Automation functionality and behavior

The automation functionality and behavior is the one of the most important aspects of the self-healing system. The automation behavior should be predictable and repeatable. The self-healing system simulator shall be capable of performing tests to completely confirm that all of these requirements are working for a particular configuration of the system.

The Department of Energy (DOE) released a study in 2014 titled “Fault Location, Isolation and Service Restoration Technologies Reduce Outage Impact” which examined different self-healing systems. They found that fully automated self-healing systems were far superior in response to automation systems that require some manual operator intervention. Furthermore, they found that utilities that deploy systems that have both operator intervention and fully automatic modes have difficulty transitioning to the fully automatic modes because of the reluctance of operations to give up partial control. Therefore, if the system is enabled, it shall run in a fully automatic capability that does not require operator intervention or review.

The self-healing system shall have the following automation behavior capabilities as standard functions.

4.1. Fault management

4.1.1. If there is a fault on the system, the self-healing automation system must detect the fault, isolate the fault to the smallest line section (zone) possible, and restore as many zones as possible

4.1.2. Fault management shall be initiated by both a fault target or fault pickup and a lockout of reclosing devices or an opening of non-reclosing devices

4.1.3. The self-healing system shall isolate the zone downstream of the furthest downstream device with a fault target and the next device that has no fault indication

4.1.4. Restoration of load to alternative feeders shall not overload any line section of the alternative feeder

4.1.5. Fault management should be capable of being enabled or disabled in the system configuration at multiple levels of the system (Global\site, subsystem, feeder, device)

4.1.6. Unfaulted zones should not be re-energized if there is any viable voltage reported on that link (above the restoration threshold)

4.1.7. Alternate feeders should not be allowed to pick up unfaulted zones if they do not have viable voltage on the link (assuming there is voltage data provided by the devices)

4.2. Voltage Management

4.2.1. If there is a Loss of Voltage (LOV) detected on the feeder, the self-healing automation system must detect the LOV and isolate the device that is the furthest upstream, and restore as many zones as possible

4.2.1.1. The system shall be capable of detecting both single phase LOV and three phase LOV

4.2.1.2. The LOV will start after the detection of the LOV and a user configurable time has expired. Alternatively, if the device opens itself and sends indication to the self-healing system to start restoration, then restoration can occur without waiting for the LOV timer

4.2.1.3. Before isolating for the LOV, the system must check to ensure that the corresponding LOV occurs on the downstream devices. If the corresponding LOV does not occur on the downstream devices, the system shall restrain the voltage loss isolation because the issue is most likely caused by a bad sensor

4.2.1.4. Restoration of load to alternative feeders shall not overload any line section of the alternative feeder

4.2.1.5. Voltage management should be capable of being enabled or disabled in the system configuration at multiple levels of the system (Global\site, subsystem, feeder, device)

4.2.1.6. Unfaulted zones should not be re-energized if there is any viable voltage reported on that link (above the restoration threshold)

4.2.1.7. Alternative feeders should not be allowed to pick up zones if they do not have viable voltage on the link (assuming there is voltage data provided by the devices)

4.2.1.8. If there is a LOV detected and the feeder has active fault targets or pickups, then the self-healing system shall treat the event as fault management and shall isolate the fault on the system. This is often the case if the most upstream fault interrupting device is not modeled in the self-healing system

4.3. Isolation switching strategies

4.3.1. The self-healing system shall attempt to isolate the fault to the smallest area possible (a single zone)

4.3.2. The self-healing system shall open as many switches as needed to isolate the faulted area of the feeder

4.3.3. If the fault cannot be isolated to a single zone due to a device in local mode, or a device without fault detection capabilities, or a device that is not communicating, then the self-healing system should attempt to isolate the fault around that limiting device, by including the device in the fault isolation area

4.3.4. The self-healing system shall confirm that the fault is isolated before attempting restoration

4.4. Default Restoration switching strategies

4.4.1. The self-healing system shall NEVER re-energize a link that has a device with a Hot-Line-Tag present on that link or zone

4.4.2. The self-healing system shall, by default, minimize the total number of switching operations needed to successfully isolate the fault and pick up the unfaulted load sections

4.4.3. If an unfaulted link (zones connected by closed switching devices) can be picked up by closing one Normally Open switching device without violating any of the load limits on that feeder, then the self-healing system should choose that Normally Open switch to restore that unfaulted link

4.4.4. If there are two or more Normally Open switches capable of restoring unfaulted zones without overloading, the self-healing system shall check to see if one is a preferred source, and if it is preferred, shall use that feeder even though it may have a lower overall capacity

4.4.5. If there are two or more Normally Open switches are capable of restoring unfaulted zones without overloading, the self-healing system shall check to see if any of those Normally Open switches will return the unfaulted load back to their original feeder and if so shall select that Normally Open switch to close so that the load is returned to its original feeder over consideration for a preferred feeder or load capacity

4.5. Load fragmentation

4.5.1. When the self-healing system determines that no one feeder can be used to restore the entire link due to load limitations, then the self-healing system will fragment the link so that multiple alternative feeders can be used to restore the unfaulted zones sections

4.5.2. If all of the unfaulted zones still cannot be restored, then the self-healing system will use zone priorities to determine which zones should be restored and which zones should be left unenergized

4.6. Alternate switching strategies

4.6.1. While the default restoration switching strategy maximizes the speed of restoration and minimizes the total number of switching commands, there are times that an alternative switching strategy may be better. The self-healing system shall have a maximum link fragmentation switching strategy that will intentionally break apart a restorable link into as many sections as can be picked up by alternative sources. This strategy will minimize the total loading that is picked up by any one feeder and therefore will be the most tolerant of daily load growth after the restoration has occurred

4.6.2. While the maximum fragmentation switching strategy is enabled, the system will disregard any preferred alternate sources for the link that may be defined

4.6.3. Maximum link fragmentation can be configured to be enabled or disabled, and if enabled in the configuration, it then can be operationally enabled or disabled by the operators

4.7. Additional attempts at restoration

4.7.1. After an event, if the self-healing system was initially unable to reconnect all of the loads due to an issue like loading, no available energized tie, or communications, then periodically thereafter, the system shall recheck to see if conditions have changed so that the unfaulted and unenergized zones can now be reconnected. The self-healing system shall have a timer that limits how long after the initial event that the unfaulted and unenergized zones can be reconnected

4.8. Load Management

4.8.1. The self-healing system shall be capable of transferring and dropping loads independent of an event condition if system conditions are such that overloads are present beyond an established time period

4.8.2. The load transfer and load drop features shall be independently enabled and disabled

4.8.3. When load drop conditions are reached, the self-healing system shall drop the lower priority load zones first if possible

4.8.4. For load transfers the self-healing system shall be able to support both “Make then Break” and “Break then Make” transfers. The system shall be able to be restricted to only allow “Make then Break” transfers

4.8.5. The self-healing system shall support multiple load limits for either seasonal loading or operated initiated load limit changes

4.8.6. The self-healing system shall support bi-directional load limits

4.8.7. The system shall have three separate alarm load level warnings

4.8.8. The self-healing system shall be capable of assessing the loading impacts of any restoration through the substation transformer

4.8.8.1. Loading information given in secondary Amps shall either be collected directly from the substation transformer sensing system or via a summation of the measurement of the first devices that are connected to the transformer bus

4.9. Simultaneous faults

4.9.1. Simultaneous faults in the same subsystem shall be handled by isolating each fault first and then sequentially attempting to restore each unfaulted link

4.9.2. Simultaneous faults in different subsystems shall be handled independently and in parallel

4.10. Miscoordinated Faults

4.10.1. At times a miscoordination may occur on a feeder causing an upstream device to operate to lockout while the fault is actually downstream of a subsequent device causing that device to indicate that a protective element “pickup” has occurred. When this type of scenario occurs, the self-healing system shall logically transfer the fault location downstream of the furthest device that saw the pick-up. It shall isolate for that fault location, and then restore both up-stream and downstream the unfaulted zones

4.10.2. This Miscoordination feature shall be capable of being enabled and disabled in the configuration

4.11. Preferred feeders

4.11.1. The self-healing system shall support preferred feeders that allow the user to configure which alternative feeder is the preferred source for unfaulted zones that need to be restored after an event

4.12. Zone Priorities

4.12.1. The self-healing system shall support zone priorities that allow the user to configure which zones are higher and lower priority in cases where not all of the unfaulted zones can be restored

4.13. Control retries

4.13.1. The self-healing system shall support a methodology to retry controls automatically if the device does not provide the proper feedback from an initial control command

4.13.2. The user shall be able to configure the number of times a control command can be resent before the self-healing system must alarm for a control command failure

4.14. Automatic Reclose Block application

4.14.1. The self-healing system shall support the automatic application of blocking the reclose capability of a device that is about to close to restore unfaulted zones

4.14.2. After the zones are successfully restored, the self-healing system shall reapply the reclosing mode of the device

4.14.3. The user shall be able to configure the automatic reclose block application as enabled or disabled at the different levels of the self-healing system

4.15. Automatic block ground trip application for Make then Break transitions

4.15.1. The self-healing system shall support the automatic application of blocking the ground trip capability of the devices that are about to form a closed loop for a Make then Break transition

4.15.2. After the Make then Break transition is complete, the self-healing system shall reapply the ground trip capabilities of the devices

4.15.3. The user shall be able to configure the automatic block ground trip application as enabled or disabled at the different levels of the self-healing system

4.16. Manage device sectionalizer modes based on topology

4.16.1. The self-healing system shall support the automatic disabling of a device sectionalizer mode based on whether the device is connected to a feeder that is different than its normal feeder based upon topology. If the device is returned to its normal feeder, the sectionalizer mode for the device shall be reenabled by the self-healing system

4.17. Prevent line transformer backfeeding

4.17.1. In situations where line transformers are used for restorations between one voltage level feeder and another, the self-healing system shall have the option to prevent the system from restoring the high voltage system from the lower voltage system through the line transformer

4.17.2. The user shall be able to configure the prevention of line transformer backfeeding at the different levels of the self-healing system

4.18. Reconnection of Distributed Generation on a return to normal command

4.18.1. The self-healing system shall have the option of either directly closing a distributed generation switching device on a return to normal command or allowing that distributed generation switching device to close itself back onto the system

4.18.2. The user shall be able to configure the Reconnection of Distributed Generation at the different levels of the self-healing system

4.19. Automatic changes to the device profile settings

4.19.1. The self-healing system shall support automatically changing the device profile settings in the event that certain system conditions are met

4.19.2. At a minimum the self-healing system shall support the recognition of these different system conditions and shall set a preconfigured profile in the IED that is user configurable

4.19.3. The system conditions that the self-healing system must be able to recognize are:

4.19.3.1. The source of the device is the normal feeder

4.19.3.2. The source of the device is an alternate feeder

4.19.3.3. The device is on a feeder that is in its normal topology

4.19.3.3.1. Before the system transitions to this state

4.19.3.3.2. After detection of this state

4.19.3.4. The device is on a feeder that has acquired a zone beyond its normal topology (the feeder is supporting more load than is normal)

4.19.3.4.1. Before the system transitions to this state

4.19.3.4.2. After detection of this state

4.19.3.5. The power flow through that device is in the forward direction based on the topology

4.19.3.5.1. Before the system transition to this state

4.19.3.5.2. After detection of this state

4.19.3.6. The power flow through that device is in the reverse direction based on the topology

4.19.3.6.1. Before the system transition to this state

4.19.3.6.2. After detection of this state

4.19.3.7. The load through that device has increased or decreased beyond a predefined user threshold

4.19.3.8. The self-healing system detects no other predefined conditions and should therefore return the device to a default setting profile

4.20. All conditions shall have user definable priorities that determine which one takes precedence when two or more conditions are present at the same time

4.21. Each condition shall have a configurable predetermined outcome if the setting profile change is not confirmed by the device

4.21.1. None: this is a “don’t care” response if the profile setting application is not confirmed by the device

4.21.2. Minor Health issue: if the device does not respond to the profile setting change control, then the self-healing system should continue to proceed with the restoration of the zone but shall alarm that the device has not correctly responded to the profile setting command

4.21.3. Major Health issue: if the device does not respond to the profile setting change control, then the self-healing system should halt further automation associated with that device and shall alarm that the device has not correctly responded to the profile setting command

4.22. In addition to changing the profile setting group for the IED on the specific condition, the self-healing system shall also be able to change the following common modes of the IED

4.22.1. Reclosing

4.22.2. Ground trip

4.22.3. Auto-sectionalizer

4.22.4. Loop scheme

4.23. Individual device automation

4.23.1. Each device that is configured in the automation system shall have the ability to be configured as automated or non-automated

4.23.2. Devices that are configured as non-automated shall provide data to the self-healing system but shall not be automatically controllable by the self-healing system. No control commands shall be sent to IEDs that are non-automated.

4.23.3. Devices that are configured as non-automated shall not count against the self-healing system device license limit

4.23.4. Devices that are configured to be automated shall be able to be set to non-automated via operator control. These devices shall count against the device license limit regardless of the present operator controlled automation state of the device

5. Operations safety and control

While the automation functionality and behavior of the self-healing system and its ability to reduce outage durations, is very important, so is its ability to safely interact with the operations crews that work on the utility power system every day. This means that the self-healing system must detect the conditions in which its functionality must be restrained to keep operations personnel safe.

The self-healing system shall be adaptable to conform to the safe working practices of the utility.

5.1. The self-healing system shall NEVER re-energize a link (a collection of zones connected by closed devices) that has a device with a Hot-Line-Tag present on that link or zone

5.2. A device that is not actively communicating with the self-healing system may have a Hot-Line-Tag present on the device that is unknown to the self-healing system, therefore, by default, the self-healing system shall consider non-communicating devices the same as devices that have an active Hot-Line-Tag, and shall not restore any link that has a device in a non-communicating state

5.2.1.1. This feature of treating a non-communicating device as a Hot-Line-Tag device may be disabled in the configuration as utilities may deploy other safety methodologies to protect personnel

5.3. The self-healing system shall be configurable to automatically disable automation after a first event has occurred

5.3.1. This feature, to automatically disable automation after a first event, shall be settable at the different levels of the system and shall define the scope of its affect by which level of the system it was set on

5.3.2. The self-healing system automation shall then be re-enabled either automatically when the power system is back in its normal state or via an operations command. This choice of re-enabling manually or when the system is in a normal state shall be configurable

5.4. The self-healing system shall be configurable to automatically disable automation after a manual operation has occurred

5.4.1. This feature, to automatically disable automation after a manual operation, shall be settable at the different levels of the system and shall define the scope of its affect by which level of the system it was set on

5.4.2. The self-healing system automation shall then be re-enabled either automatically when the power system is back in its normal state or via an operations command. This choice of re-enabling manually or when the system is in a normal state shall be configurable

5.5. Operators shall be able to either enable or disable automation at each level of the system (Global-Site, Subsystem, Feeder, and Device)

5.6. Field personnel should be able to disable automation on a feeder (Feeder Block) by setting one or more of the following modes of a device. The configuration of the system will need to be set for this action to occur. Once the condition that caused this feeder disable is no longer present, the self-healing system shall be allowed to resume automation of that feeder

5.6.1. Hot-Line-Tag

5.6.2. Local Mode

5.6.3. Communications failure

5.6.4. Reclose block

5.6.5. Other (any binary point or combination thereof from the device can be used to block automation on the feeder)

5.7. Field personnel should be able to disable automation on a subsystem (Subsystem Block) by setting one or more of the following modes of a device. The configuration of the system will need to be set for this action to occur. Once the condition that caused this subsystem disable is no longer present, the self-healing system shall be allowed to resume automation of that subsystem

5.7.1. Hot-Line-Tag

5.7.2. Local Mode

5.7.3. Communications failure

5.7.4. Reclose block

5.7.5. Other (any binary point or combination thereof from the device can be used to block automation on the subsystem)

5.8. Operators shall be able to set a device in the system as “bypassed” to mimic that state in the field. Closed devices will be able to be bypassed and the self-healing system will ignore all statuses from a device in the bypassed state as if it were replaced with a line and the two zones on either side of the device will be merged into one logical zone. The self-healing system will not respond to any communication, lack thereof, or data from a bypassed device including a Hot-Line-Tag indication

5.9. Operators shall be able to set a device in the system as “out of service” to mimic that state in the field. Open devices will be able to be set Out of Service and the self-healing system will ignore all statuses from a device in the Out of Service state as if it did not exist and the two zones on either side of it were separated by an air gap. The self-healing system will not respond to any communication, lack thereof, or data from an Out of Service device including a Hot-Line-Tag indication

6. Security

Security of the self-healing system is vitally important to ensure that only the correct individuals have access and control. The following requirements shall be met to ensure system security.

6.1. Authentication and Authorization

6.1.1. The self-healing system shall use the operating system authentication

6.1.1.1. No user account passwords shall be stored by the self-healing system

6.1.1.2. The user shall not be required to enter a password while accessing the system; instead the system shall verify that the user has authorization based on the access privileges for their account within the operating system

6.1.2. Self-healing system user interfaces shall have authorization controls

6.1.3. Only Administrators defined within the self-healing system shall have the ability to change authorization privileges of individuals or groups within the self-healing system. Operating system administrators shall not have this right by default. The self-healing administration right must be specifically assigned

6.1.3.1. The user that installs the self-healing software on the server shall be the initially authorized administrator within the self-healing system

6.1.4. Account authorization shall be configured to only allow the required access for the specific job types

6.1.5. Concurrent logins by the same user shall be blocked

6.1.6. Visualization of other user active sessions shall be available to authorized users

6.1.7. The self-healing system shall provide users with the ability to see their last logon date and time

6.2. Logging

6.2.1. The self-healing system shall generate logs for all user activity

6.2.2. The self-healing system shall generate logs for all component activity

6.2.3. Each log shall have:

6.2.3.1. Date and time information

6.2.3.2. Description

6.2.3.3. Level

6.2.3.3.1. Error

6.2.3.3.2. Warning

6.2.3.3.3. Informational

6.2.3.3.4. Debug

6.2.3.3.5. Trace

6.2.4. The self-healing system shall have informational, warning and error logs enabled by default with the ability to change the logging level if needed

6.2.5. Logs shall be stored locally and shall support a syslog remote archiving process

6.2.6. Local logs shall have an encryption methodology leveraged through the operating system

6.3. Notification

6.3.1. The self-healing system shall be capable of sending alerts via email (with proper access to an email server) for process level events

6.4. System hardening

6.4.1. The self-healing system shall have a system hardening procedure

6.4.2. The system hardening shall include information on establishment of process accounts for the self-healing system

6.4.3. Process accounts for the self-healing system shall have least required system privileges established

6.5. Backup and restore

6.5.1. The self-healing system shall be configurable to automatically back-up all system configurations and shall have a method for restoring to a previous configuration if the need arises

6.5.2. The configuration backups shall occur on a period basis if configured to do so. The user shall be able to manually trigger a backup process as well.

6.6. Operating system update review

6.6.1. The self-healing system shall have a formal operating system patch test process and user notification process if such tests conclude that the OS patch has caused self-healing system operational performance degradation.

6.7. Software updating

6.7.1. New software shall be available via a website for clients so that they may update when it is best for their systems

6.7.2. Release notes shall be generated with each release of the software

6.7.3. Each release shall be tested so that prior functionality has not been negatively impacted.

6.7.4. Clients shall be notified when new software is available for download

6.8. Remote client control limitations

6.8.1. The self-healing system shall have the ability to limit operational functionality of authorized users that are accessing via remote clients as opposed to a local terminal operational functionality

6.8.2. The self-healing system shall display which server, the remote clients are currently connected to

6.8.3. The self-healing system shall have a settable session timeout length

7. Server and network requirements

The self-healing system shall operate from a server having the following minimum requirements:

Operating Systems	Windows 10, Windows Server 2016, Windows Server 2019 (recommended)
Operating System Platform	x64
Memory Speed	2667 MHz
Memory Capacity	16 Gigabyte base memory. Add 1 Gigabyte of memory for each additional FAM region with less than 300 devices. Add 1 Gigabyte of memory for each additional 300 devices in a region
CPU Speed	2.6 Gigahertz Intel® Compatible
Processor Cores	4 processor cores for the base system. Add 1 processor core for each additional FAM region with less than 300 devices. Add 1 processor core for each additional 300 devices in a region
Server Drive Size	500 Gigabytes
Server Drive Type	SATA for better performance or RAID if data persistence is required.

Network Adapter	Dual Gigabit Network Adapter

7.1. Memory size and CPU Speed may be increased for better performance. Increasing CPU speed will provide more improvement than increasing the number of CPUs. Windows Server systems may require “Application Server” role added for installation to complete. This role will correct an error during installation about requiring .NET 3.5 Framework.

7.2. The self-healing system shall be capable of running on a virtual machine (VM) of the proper operating system

8. Redundancy

8.1. The self-healing system shall support redundancy of the central processing unit

8.2. The redundant system shall be a complete fully-functional standby unit capable of replicating all functions of the primary device in a failover situation

8.3. Redundant hardware components shall support redundant links for the sharing of health status and automation function modes in real time

8.4. The redundant pair must support a shared virtual IP address such that external devices may interrogate both automation processors without any additional configuration or equipment

9. Licensing

9.1. The self-healing system shall support both a hardware HASP (USB) or a software HASP licensing mechanism

9.2. Attributes applied to the HASP which enable features in the self-healing software shall be upgradable via a file that can be electronically transmitted to clients

9.3. License attributes shall be viewable once applied

9.4. Conflicts with the licensing shall be viewable in the logs

10. Visualization

The self-healing software must provide client applications for managing the following aspects of the system:

10.1. A primary interface to manage security, users, licensing, redundancy, configurations, file management, and other client applications

10.2. An interface to visualize the topology and provide direct user operation and control

10.3. An interface for system diagnostics and direct visualization of the real-time database

10.4. An interface for configuring the settings that will govern the behavior of the automation system

10.5. An interface for configuration of the communications system and data gathering protocols

10.6. An interface that provides a mean to view and sort the logs generated by the system

11. Simulation

11.1. The automation software must provide a simulator capable of producing the data from the intelligent electronic devices (IED) and proper response of the data from those IEDs back to the self-healing system

in such a way that the self-healing automation logic is reacting to those data changes as if they were coming directly from the IEDs.

11.1.1. The automation engine portion of the self-healing system shall not be aware that it is in a simulation mode. This way the reactions of the self-healing system to the data changes from the simulator will provoke the same reaction from actual IEDs in production

11.2. The simulator shall integrate with the human machine interface (HMI) such that most system events may be initiated from the graphical diagram

11.3. Simulation shall include the ability to test fault and voltage events, frequency events, operating modes of protective devices, communication failures, communication delays, operating modes of the automation system, mechanism failures, and system loading

11.4. The simulator shall be able to operate in real time mode and fast simulation mode

11.5. The simulation capabilities shall not be dependent on any actual physical equipment

11.6. The simulation shall be able to interact with lab equipment if so desired

12. Configurable

12.1. The self-healing system shall be programmed or configured by the end user in a way that requires no logic equations, computer language programming, or scripting

12.2. The automation software shall supply a Windows-menu formatted configurable tool for defining the system topology and behavior

12.3. All configuration files associated with the automation solution shall be transferred between computing platforms in a single file