

# Automating and Customizing the Node Health Check Tool for Support System Application

Aedan Wells | HPC-SYS | New Mexico Tech | aedan.wells@student.nmt.edu

## 1. Motivation

Supporting systems within a high-performance computing ecosystem need to be stable for researchers. However, all production machines have the possibility to run into errors that can. Node Health Check (NHC) from Lawrence Berkely National Labs works well for solving this but application to many machines is not simple to apply or correct specific errors. This project developed machine customizable ansible roles for NHC application and custom checks for specific errors.

## 2. Node Health Check

NHC is a tool that runs specific check functions written in bash to enable wholistic monitoring for system health. Some built in checks include:

- Filesystem Free Space
- Process Daemon activity
- System specific check

See Figure 1 for the flow chart for the Node Health Check process

Node Health Check is **adaptable** and **expandable** to be applied **uniquely** to any machine for **automated** health monitoring

## 3. Machine Application

NHC is applied to machines via an ansible role that takes in the specific machine rules in the platforms inventory yaml and creates an appropriate configuration for them. This allows for:

- Default application of checks valid for all machines
- Application of specific checks only valid for 1 machine

## 4. Custom Checks

Node Health check includes many simple check functions; however, these are harder to adapt to detect/prevent common errors. This project entailed creating the following custom check functions that check:

- Bonding health
- Iptable proper application
- Raid disk health
- RHEL subscriptions
- Root password encryption
- Smart disk health and testing
- Docker specific iptable rule status

These allow for a better detection for common issues on Support System machines

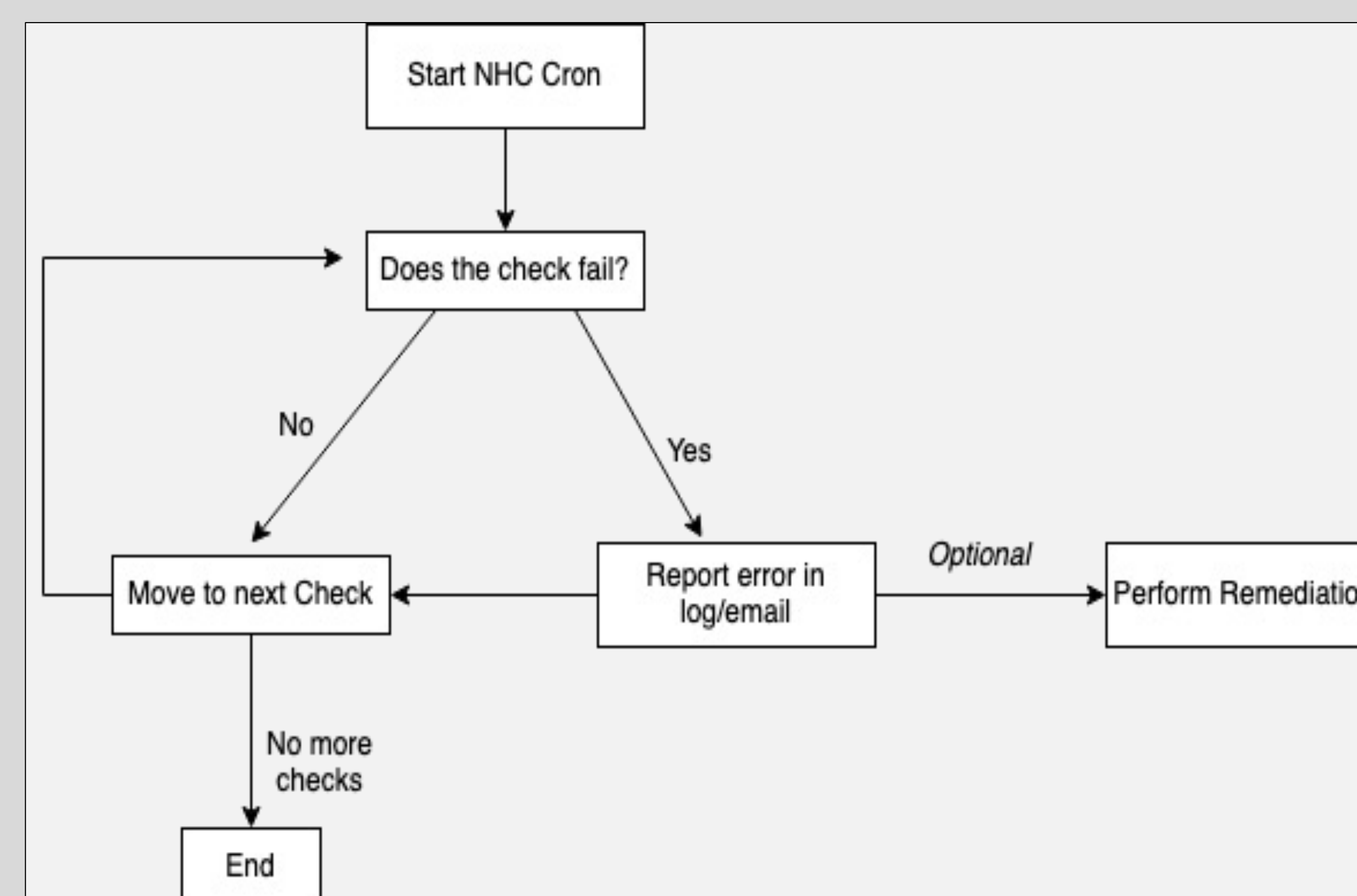


Figure 1: NHC Process Flowchart

## Acknowledgements

- Mentors: Jonathan Nielsen, Michael Jennings
- Support Systems: Conor Robinson, Nick Jones