

## SUDO CALLS LIST

The following table lists available `sudo` commands along with their purpose and whether they can accept wildcards.

Command	Wildcards	Purpose
<code>/bin/chown -R &lt;user&gt; /opt/apps/verodin/node/node</code>	FALSE	Upgrade Handling
<code>/bin/netstat -anp   grep {process name}</code>	TRUE	Network Control
<code>/bin/systemctl daemon-reload</code>	FALSE	Service Control
<code>/bin/systemctl disable chronyd</code>	FALSE	NTP Control
<code>/bin/systemctl enable chronyd</code>	FALSE	NTP Control
<code>/bin/systemctl mask nginx</code>	FALSE	Service Control
<code>/bin/systemctl mask sshd</code>	FALSE	Service Control
<code>/bin/systemctl mask verodin-backend</code>	FALSE	Service Control
<code>/bin/systemctl mask verodin-network-monitor</code>	FALSE	Service Control
<code>/bin/systemctl mask verodin-pull</code>	FALSE	Service Control
<code>/bin/systemctl mask verodin-web</code>	FALSE	Service Control
<code>/bin/systemctl reload nginx</code>	FALSE	Service Control
<code>/bin/systemctl restart chronyd</code>	FALSE	NTP Control
<code>/bin/systemctl restart nginx</code>	FALSE	Service Control
<code>/bin/systemctl restart verodin-backend</code>	FALSE	Service Control
<code>/bin/systemctl restart verodin-network-monitor</code>	FALSE	Service Control
<code>/bin/systemctl restart verodin-pull</code>	FALSE	Service Control
<code>/bin/systemctl restart verodin-web</code>	FALSE	Service Control
<code>/bin/systemctl start ssh-second.service</code>	FALSE	SSH Tunnel Control
<code>/bin/systemctl start sshd</code>	FALSE	Service Control
<code>/bin/systemctl stop chronyd</code>	FALSE	NTP Control
<code>/bin/systemctl stop nginx</code>	FALSE	Service Control
<code>/bin/systemctl stop ssh-second.service</code>	FALSE	SSH Tunnel Control
<code>/bin/systemctl stop sshd</code>	FALSE	Service Control
<code>/bin/systemctl stop verodin-backend</code>	FALSE	Service Control

Command	Wildcards	Purpose
<code>/bin/systemctl stop verodin-network-monitor</code>	FALSE	Service Control
<code>/bin/systemctl stop verodin-pull</code>	FALSE	Service Control
<code>/bin/systemctl stop verodin-web</code>	FALSE	Service Control
<code>/bin/systemctl unmask nginx</code>	FALSE	Service Control
<code>/bin/systemctl unmask sshd</code>	FALSE	Service Control
<code>/bin/systemctl unmask verodin-backend</code>	FALSE	Service Control
<code>/bin/systemctl unmask verodin-network-monitor</code>	FALSE	Service Control
<code>/bin/systemctl unmask verodin-pull</code>	FALSE	Service Control
<code>/bin/systemctl unmask verodin-web</code>	FALSE	Service Control
<code>/opt/apps/verodin/node/ext/hans-master/hans</code>	TRUE	ICMP Tunnel
<code>/opt/apps/verodin/node/ext/iodine/bin/iodine</code>	TRUE	DNS Tunnel
<code>/sbin/iptables -A INPUT -i lo -p tcp --dport &lt;port&gt; -j ACCEPT</code>	FALSE	open loopback port
<code>/sbin/iptables -A INPUT -p {proto} -m multiport --dports {portstr} -m state --state NEW,ESTABLISHED -j ACCEPT</code>	TRUE	Firewall Control
<code>/sbin/iptables -A INPUT -p {protostr} --dport {portstr} -m state --state NEW,ESTABLISHED -j ACCEPT</code>	TRUE	Firewall Control
<code>/sbin/iptables -A INPUT -s {ipstr}/32 -p {protostr} --dport {portstr} -m state --state NEW,ESTABLISHED -j ACCEPT</code>	TRUE	Firewall Control
<code>/sbin/iptables -D INPUT -i lo -p tcp --dport &lt;port&gt; -j ACCEPT</code>	FALSE	close loopback port
<code>/sbin/iptables -D INPUT -p {proto} -m multiport --dports {portstr} -m state --state NEW,ESTABLISHED -j ACCEPT</code>	TRUE	Firewall Control
<code>/sbin/iptables -D INPUT -p {protostr} --dport {portstr} -m state --state NEW,ESTABLISHED -j ACCEPT</code>	TRUE	Firewall Control
<code>/sbin/iptables -D INPUT -s {ipstr}/32 -p {protostr} --dport {portstr} -m state --state NEW,ESTABLISHED -j ACCEPT</code>	TRUE	Firewall Control
<code>/sbin/iptables-restore &lt; /opt/apps/verodin/node/node/tmp/iptables/update.txt</code>	FALSE	Firewall Control
<code>/sbin/setcap CAP_NET_BIND_SERVICE=+eip /opt/apps/verodin/node/node/scripts/verodin_backend_service</code>	FALSE	Upgrade Handling
<code>/usr/bin/date -s {sign} {adjust} seconds</code>	TRUE	NTP Control
<code>/usr/bin/sysctl -w net.ipv4.ip_local_reserved_ports=""</code>	FALSE	Network Control
<code>/usr/bin/sysctl -w net.ipv4.ip_local_reserved_ports="{ports}"</code>	TRUE	Network Control
<code>/usr/bin/sysctl -w net.ipv4.tcp_orphan_retries=0</code>	FALSE	Action tcp orphan retry disable

Command	Wildcards	Purpose
<code>/usr/bin/sysctl -w net.ipv4.tcp_orphan_retries=1</code>	FALSE	Action tcp orphan retry enable
<code>/usr/sbin/ifdown &lt;nic&gt;</code>	TRUE	Interface Control
<code>/usr/sbin/ifup &lt;nic&gt;</code>	TRUE	Interface Control
<code>/usr/sbin/ip link set {nic} down</code>	TRUE	Interface Control
<code>/usr/sbin/ip link set {nic} up</code>	TRUE	Interface Control
<code>/usr/sbin/ntpdate &lt;ntphost&gt;</code>	TRUE	NTP Control
<code>/usr/sbin/shutdown -r now</code>	FALSE	Shutdown
<code>nmcli con reload {interface}</code>	TRUE	Interface Route Handling
<code>nmcli connection modify {interface} -ipv4.routes "{destination}/{cidr} {gateway}"</code>	TRUE	Interface Route Handling
<code>nmcli connection modify {interface} +ipv4.routes "{destination}/{cidr} {gateway}"</code>	TRUE	Interface Route Handling
<code>nmcli connection modify {interface} ipv4.route-metric {metric}</code>	TRUE	Interface Route Handling
<code>nmcli connection up {interface}</code>	TRUE	Interface Route Handling
<code>timedatectl set-ntp 0</code>	FALSE	NTP Control
<code>timedatectl set-ntp 1</code>	FALSE	NTP Control
<code>touch /opt/apps/verodin/node/node/settings.json</code>	FALSE	Actor Settings handling