

## Nagios Standard Macros

Host Macros.....	1
Host Group Macros.....	4
Service Macros.....	4
Service Group Macros.....	7
Contact Macros.....	7
Contact Group Macros.....	8
Summary Macros.....	8
Notification Macros.....	9
Date/Time Macros.....	10
File Macros.....	11
Misc Macros.....	11

### Host Macros

\$HOSTNAMES\$	Short name for the host (i.e. "biglinuxbox"). This value is taken from the <i>host_name</i> directive in the <a href="#">host definition</a> .
\$HOSTDISPLAYNAMES\$	An alternate display name for the host. This value is taken from the <i>display_name</i> directive in the <a href="#">host definition</a> .
\$HOSTALIASS\$	Long name/description for the host. This value is taken from the <i>alias</i> directive in the <a href="#">host definition</a> .
\$HOSTADDRESS\$	Address of the host. This value is taken from the <i>address</i> directive in the <a href="#">host definition</a> .
\$HOSTSTATES\$	A string indicating the current state of the host ("UP", "DOWN", or "UNREACHABLE").
\$HOSTSTATEIDS\$	A number that corresponds to the current state of the host: 0=UP, 1=DOWN, 2=UNREACHABLE.
\$LASTHOSTSTATES\$	A string indicating the last state of the host ("UP", "DOWN", or "UNREACHABLE").
\$LASTHOSTSTATEIDS\$	A number that corresponds to the last state of the host: 0=UP, 1=DOWN, 2=UNREACHABLE.
\$HOSTSTATETYPES\$	A string indicating the <a href="#">state type</a> for the current host check ("HARD" or "SOFT"). Soft states occur when host checks return a non-OK (non-UP) state and are in the process of being retried. Hard states result when host checks have been checked a specified maximum number of times.
\$HOSTATTEMPTS\$	The number of the current host check retry. For instance, if this is the second time that the host is being rechecked, this will be the number two. Current attempt number is really only useful when writing host event handlers for "soft" states that take a specific action based on the host retry number.
\$MAXHOSTATTEMPTS\$	The max check attempts as defined for the current host. Useful when writing host event handlers for "soft" states that take a specific action based on the host retry number.
\$HOSTEVENTIDS\$	A globally unique number associated with the host's current state. Every time a host (or service) experiences a state change, a global event ID

	number is incremented by one (1). If a host has experienced no state changes, this macro will be set to zero (0).
\$LASTHOSTEVENTID\$	The previous (globally unique) event number that was given to the host.
\$HOSTPROBLEMID\$	A globally unique number associated with the host's current problem state. Every time a host (or service) transitions from an UP or OK state to a problem state, a global problem ID number is incremented by one (1). This macro will be non-zero if the host is currently a non-UP state. State transitions between non-UP states (e.g. DOWN to UNREACHABLE) do not cause this problem id to increase. If the host is currently in an UP state, this macro will be set to zero (0). Combined with event handlers, this macro could be used to automatically open trouble tickets when hosts first enter a problem state.
\$LASTHOSTPROBLEMID\$	The previous (globally unique) problem number that was given to the host. Combined with event handlers, this macro could be used for automatically closing trouble tickets, etc. when a host recovers to an UP state.
\$HOSTLATENCY\$	A (floating point) number indicating the number of seconds that a <i>scheduled</i> host check lagged behind its scheduled check time. For instance, if a check was scheduled for 03:14:15 and it didn't get executed until 03:14:17, there would be a check latency of 2.0 seconds. On-demand host checks have a latency of zero seconds.
\$HOSTEXECUTIONTIMES\$	A (floating point) number indicating the number of seconds that the host check took to execute (i.e. the amount of time the check was executing).
\$HOSTDURATIONS\$	A string indicating the amount of time that the host has spent in its current state. Format is "XXh YYm ZZs", indicating hours, minutes and seconds.
\$HOSTDURATIONSECS\$	A number indicating the number of seconds that the host has spent in its current state.
\$HOSTDOWNTIMES\$	A number indicating the current "downtime depth" for the host. If this host is currently in a period of <a href="#">scheduled downtime</a> , the value will be greater than zero. If the host is not currently in a period of downtime, this value will be zero.
\$HOSTPERCENTCHANGES\$	A (floating point) number indicating the percent state change the host has undergone. Percent state change is used by the <a href="#">flap detection</a> algorithm.
\$HOSTGROUPNAMES\$	The short name of the hostgroup that this host belongs to. This value is taken from the <i>hostgroup_name</i> directive in the <a href="#">hostgroup definition</a> . If the host belongs to more than one hostgroup this macro will contain the name of just one of them.
\$HOSTGROUPNAMES\$	A comma separated list of the short names of all the hostgroups that this host belongs to.
\$LASTHOSTCHECK\$	This is a timestamp in time_t format (seconds since the UNIX epoch) indicating the time at which a check of the host was last performed.
\$LASTHOSTSTATECHANGES\$	This is a timestamp in time_t format (seconds since the UNIX epoch) indicating the time the host last changed state.
\$LASTHOSTUPS\$	This is a timestamp in time_t format (seconds since the UNIX epoch) indicating the time at which the host was last detected as being in an UP state.

\$LASTHOSTDOWN\$	This is a timestamp in time_t format (seconds since the UNIX epoch) indicating the time at which the host was last detected as being in a DOWN state.
\$LASTHOSTUNREACHABLE\$	This is a timestamp in time_t format (seconds since the UNIX epoch) indicating the time at which the host was last detected as being in an UNREACHABLE state.
\$HOSTOUTPUT\$	The first line of text output from the last host check (i.e. "Ping OK").
\$LONGHOSTOUTPUT\$	The full text output (aside from the first line) from the last host check.
\$HOSTPERFDATA\$	This macro contains any <a href="#">performance data</a> that may have been returned by the last host check.
\$HOSTCHECKCOMMAND\$	This macro contains the name of the command (along with any arguments passed to it) used to perform the host check.
\$HOSTACKAUTHOR\$	A string containing the name of the user who acknowledged the host problem. This macro is only valid in notifications where the \$NOTIFICATIONTYPE\$ macro is set to "ACKNOWLEDGEMENT".
\$HOSTACKAUTHORNAME\$	A string containing the short name of the contact (if applicable) who acknowledged the host problem. This macro is only valid in notifications where the \$NOTIFICATIONTYPE\$ macro is set to "ACKNOWLEDGEMENT".
\$HOSTACKAUTHORALIAS\$	A string containing the alias of the contact (if applicable) who acknowledged the host problem. This macro is only valid in notifications where the \$NOTIFICATIONTYPE\$ macro is set to "ACKNOWLEDGEMENT".
\$HOSTACKCOMMENT\$	A string containing the acknowledgement comment that was entered by the user who acknowledged the host problem. This macro is only valid in notifications where the \$NOTIFICATIONTYPE\$ macro is set to "ACKNOWLEDGEMENT".
\$HOSTACTIONURL\$	Action URL for the host. This macro may contain other macros (e.g. \$HOSTNAME\$), which can be useful when you want to pass the host name to a web page.
\$HOSTNOTESURL\$	Notes URL for the host. This macro may contain other macros (e.g. \$HOSTNAME\$), which can be useful when you want to pass the host name to a web page.
\$HOSTNOTES\$	Notes for the host. This macro may contain other macros (e.g. \$HOSTNAME\$), which can be useful when you want to host-specific status information, etc. in the description.
\$TOTALHOSTSERVICES\$	The total number of services associated with the host.
\$TOTALHOSTSERVICESOK\$	The total number of services associated with the host that are in an OK state.
\$TOTALHOSTSERVICESWARNING\$	The total number of services associated with the host that are in a WARNING state.
\$TOTALHOSTSERVICESUNKNOWN\$	The total number of services associated with the host that are in an UNKNOWN state.
\$TOTALHOSTSERVICESCRITICAL\$	The total number of services associated with the host that are in a CRITICAL state.

## Host Group Macros

\$HOSTGROUPALIAS\$	The long name / alias of either 1) the hostgroup name passed as an on-demand macro argument or 2) the primary hostgroup associated with the current host (if not used in the context of an on-demand macro). This value is taken from the <i>alias</i> directive in the <a href="#">hostgroup definition</a> .
\$HOSTGROUPMEMBERS\$	A comma-separated list of all hosts that belong to either 1) the hostgroup name passed as an on-demand macro argument or 2) the primary hostgroup associated with the current host (if not used in the context of an on-demand macro).
\$HOSTGROUPNOTES\$	The notes associated with either 1) the hostgroup name passed as an on-demand macro argument or 2) the primary hostgroup associated with the current host (if not used in the context of an on-demand macro). This value is taken from the <i>notes</i> directive in the <a href="#">hostgroup definition</a> .
\$HOSTGROUPNOTESURL\$	The notes URL associated with either 1) the hostgroup name passed as an on-demand macro argument or 2) the primary hostgroup associated with the current host (if not used in the context of an on-demand macro). This value is taken from the <i>notes_url</i> directive in the <a href="#">hostgroup definition</a> .
\$HOSTGROUPACTIONURL\$	The action URL associated with either 1) the hostgroup name passed as an on-demand macro argument or 2) the primary hostgroup associated with the current host (if not used in the context of an on-demand macro). This value is taken from the <i>action_url</i> directive in the <a href="#">hostgroup definition</a> .

## Service Macros

\$SERVICEDESC\$	The long name/description of the service (i.e. "Main Website"). This value is taken from the <i>service_description</i> directive of the <a href="#">service definition</a> .
\$SERVICEDISPLAYNAME\$	An alternate display name for the service. This value is taken from the <i>display_name</i> directive in the <a href="#">service definition</a> .
\$SERVICESTATES\$	A string indicating the current state of the service ("OK", "WARNING", "UNKNOWN", or "CRITICAL").
\$SERVICESTATEID\$	A number that corresponds to the current state of the service: 0=OK, 1=WARNING, 2=CRITICAL, 3=UNKNOWN.
\$LASTSERVICESTATES\$	A string indicating the last state of the service ("OK", "WARNING", "UNKNOWN", or "CRITICAL").
\$LASTSERVICESTATEID\$	A number that corresponds to the last state of the service: 0=OK, 1=WARNING, 2=CRITICAL, 3=UNKNOWN.
\$SERVICESTATETYPES\$	A string indicating the <a href="#">state type</a> for the current service check ("HARD" or "SOFT"). Soft states occur when service checks return a non-OK state and are in the process of being retried. Hard states result when service checks have been checked a specified maximum number of times.
\$SERVICEATTEMPTS\$	The number of the current service check retry. For instance, if this is the second time that the service is being rechecked, this will be the number two. Current attempt number is really only useful when writing service event handlers for "soft" states that take a specific action based on the service retry number.

\$MAXSERVICEATTEMPTS\$	The max check attempts as defined for the current service. Useful when writing host event handlers for "soft" states that take a specific action based on the service retry number.
\$SERVICEISVOLATILE\$	Indicates whether the service is marked as being volatile or not: 0 = not volatile, 1 = volatile.
\$SERVICEEVENTID\$	A globally unique number associated with the service's current state. Every time a service (or host) experiences a state change, a global event ID number is incremented by one (1). If a service has experienced no state changes, this macro will be set to zero (0).
\$LASTSERVICEEVENTIDS\$	The previous (globally unique) event number that given to the service.
\$SERVICEPROBLEMIDS\$	A globally unique number associated with the service's current problem state. Every time a service (or host) transitions from an OK or UP state to a problem state, a global problem ID number is incremented by one (1). This macro will be non-zero if the service is currently a non-OK state. State transitions between non-OK states (e.g. WARNING to CRITICAL) do not cause this problem id to increase. If the service is currently in an OK state, this macro will be set to zero (0). Combined with event handlers, this macro could be used to automatically open trouble tickets when services first enter a problem state.
\$LASTSERVICEPROBLEMIDS\$	The previous (globally unique) problem number that was given to the service. Combined with event handlers, this macro could be used for automatically closing trouble tickets, etc. when a service recovers to an OK state.
\$SERVICELATENCY\$	A (floating point) number indicating the number of seconds that a scheduled service check lagged behind its scheduled check time. For instance, if a check was scheduled for 03:14:15 and it didn't get executed until 03:14:17, there would be a check latency of 2.0 seconds.
\$SERVICEEXECUTIONTIMES\$	A (floating point) number indicating the number of seconds that the service check took to execute (i.e. the amount of time the check was executing).
\$SERVICEDURATION\$	A string indicating the amount of time that the service has spent in its current state. Format is "XXh YYm ZZs", indicating hours, minutes and seconds.
\$SERVICEDURATIONSEC\$	A number indicating the number of seconds that the service has spent in its current state.
\$SERVICEDOWNTIMES\$	A number indicating the current "downtime depth" for the service. If this service is currently in a period of <a href="#">scheduled downtime</a> , the value will be greater than zero. If the service is not currently in a period of downtime, this value will be zero.
\$SERVICEPERCENTCHANGES\$	A (floating point) number indicating the percent state change the service has undergone. Percent state change is used by the <a href="#">flap detection</a> algorithm.
\$SERVICEGROUPNAME\$	The short name of the servicegroup that this service belongs to. This value is taken from the <i>servicegroup_name</i> directive in the <a href="#">servicegroup</a> definition. If the service belongs to more than one servicegroup this macro will contain the name of just one of them.
\$SERVICEGROUPNAMESS\$	A comma separated list of the short names of all the servicegroups that this service belongs to.
\$LASTSERVICECHECK\$	This is a timestamp in time_t format (seconds since the UNIX epoch) indicating the time at which a check of the service was last performed.
\$LASTSERVICESTATECHANGES\$	This is a timestamp in time_t format (seconds since the UNIX epoch)

	indicating the time the service last changed state.
\$LASTSERVICEOK\$	This is a timestamp in time_t format (seconds since the UNIX epoch) indicating the time at which the service was last detected as being in an OK state.
\$LASTSERVICEWARNING\$	This is a timestamp in time_t format (seconds since the UNIX epoch) indicating the time at which the service was last detected as being in a WARNING state.
\$LASTSERVICEUNKNOWN\$	This is a timestamp in time_t format (seconds since the UNIX epoch) indicating the time at which the service was last detected as being in an UNKNOWN state.
\$LASTSERVICECRITICAL\$	This is a timestamp in time_t format (seconds since the UNIX epoch) indicating the time at which the service was last detected as being in a CRITICAL state.
\$SERVICEOUTPUT\$	The first line of text output from the last service check (i.e. "Ping OK").
\$LONGSERVICEOUTPUT\$	The full text output (aside from the first line) from the last service check.
\$SERVICEPERFDATA\$	This macro contains any <a href="#">performance data</a> that may have been returned by the last service check.
\$SERVICECHECKCOMMAND\$	This macro contains the name of the command (along with any arguments passed to it) used to perform the service check.
\$SERVICEACKAUTHOR\$	A string containing the name of the user who acknowledged the service problem. This macro is only valid in notifications where the \$NOTIFICATIONTYPE\$ macro is set to "ACKNOWLEDGEMENT".
\$SERVICEACKAUTHORNAME\$	A string containing the short name of the contact (if applicable) who acknowledged the service problem. This macro is only valid in notifications where the \$NOTIFICATIONTYPE\$ macro is set to "ACKNOWLEDGEMENT".
\$SERVICEACKAUTHORALIAS\$	A string containing the alias of the contact (if applicable) who acknowledged the service problem. This macro is only valid in notifications where the \$NOTIFICATIONTYPE\$ macro is set to "ACKNOWLEDGEMENT".
\$SERVICEACKCOMMENT\$	A string containing the acknowledgement comment that was entered by the user who acknowledged the service problem. This macro is only valid in notifications where the \$NOTIFICATIONTYPE\$ macro is set to "ACKNOWLEDGEMENT".
\$SERVICEACTIONURL\$	Action URL for the service. This macro may contain other macros (e.g. \$HOSTNAME\$ or \$SERVICEDESC\$), which can be useful when you want to pass the service name to a web page.
\$SERVICENOTESURL\$	Notes URL for the service. This macro may contain other macros (e.g. \$HOSTNAME\$ or \$SERVICEDESC\$), which can be useful when you want to pass the service name to a web page.
\$SERVICENOTES\$	Notes for the service. This macro may contain other macros (e.g. \$HOSTNAME\$ or \$SERVICESTATE\$), which can be useful when you want to service-specific status information, etc. in the description

## Service Group Macros

\$SERVICEGROUPALIAS\$	The long name / alias of either 1) the servicegroup name passed as an on-demand macro argument or 2) the primary servicegroup associated with the current service (if not used in the context of an on-demand macro). This value is taken from the <i>alias</i> directive in the <a href="#">servicegroup definition</a> .
\$SERVICEGROUPMEMBERS\$	A comma-separated list of all services that belong to either 1) the servicegroup name passed as an on-demand macro argument or 2) the primary servicegroup associated with the current service (if not used in the context of an on-demand macro).
\$SERVICEGROUPNOTES\$	The notes associated with either 1) the servicegroup name passed as an on-demand macro argument or 2) the primary servicegroup associated with the current service (if not used in the context of an on-demand macro). This value is taken from the <i>notes</i> directive in the <a href="#">servicegroup definition</a> .
\$SERVICEGROUPNOTESURL\$	The notes URL associated with either 1) the servicegroup name passed as an on-demand macro argument or 2) the primary servicegroup associated with the current service (if not used in the context of an on-demand macro). This value is taken from the <i>notes_url</i> directive in the <a href="#">servicegroup definition</a> .
\$SERVICEGROUPNOTES\$	The action URL associated with either 1) the servicegroup name passed as an on-demand macro argument or 2) the primary servicegroup associated with the current service (if not used in the context of an on-demand macro). This value is taken from the <i>action_url</i> directive in the <a href="#">servicegroup definition</a> .

## Contact Macros

\$CONTACTNAME\$	Short name for the contact (i.e. "jdoe") that is being notified of a host or service problem. This value is taken from the <i>contact_name</i> directive in the <a href="#">contact definition</a> .
\$CONTACTALIAS\$	Long name/description for the contact (i.e. "John Doe") being notified. This value is taken from the <i>alias</i> directive in the <a href="#">contact definition</a> .
\$CONTACTEMAIL\$	Email address of the contact being notified. This value is taken from the <i>email</i> directive in the <a href="#">contact definition</a> .
\$CONTACTPAGER\$	Pager number/address of the contact being notified. This value is taken from the <i>pager</i> directive in the <a href="#">contact definition</a> .
\$CONTACTADDRESSn\$	Address of the contact being notified. Each contact can have six different addresses (in addition to email address and pager number). The macros for these addresses are \$CONTACTADDRESS1\$ - \$CONTACTADDRESS6\$. This value is taken from the <i>addressx</i> directive in the <a href="#">contact definition</a> .
\$CONTACTGROUPNAME\$	The short name of the contactgroup that this contact is a member of. This value is taken from the <i>contactgroup_name</i> directive in the <a href="#">contactgroup definition</a> . If the contact belongs to more than one contactgroup this macro will contain the name of just one of them.
\$CONTACTGROUPNAME\$	A comma separated list of the short names of all the contactgroups that this contact is a member of.

## Contact Group Macros

\$CONTACTGROUPALIAS\$	The long name / alias of either 1) the contactgroup name passed as an on-demand macro argument or 2) the primary contactgroup associated with the current contact (if not used in the context of an on-demand macro). This value is taken from the <i>alias</i> directive in the <a href="#">contactgroup definition</a> .
\$CONTACTGROUPMEMBERS\$	A comma-separated list of all contacts that belong to either 1) the contactgroup name passed as an on-demand macro argument or 2) the primary contactgroup associated with the current contact (if not used in the context of an on-demand macro).

## Summary Macros

\$TOTALHOSTSUP\$	This macro reflects the total number of hosts that are currently in an UP state.
\$TOTALHOSTSDOWN\$	This macro reflects the total number of hosts that are currently in a DOWN state.
\$TOTALHOSTSUNREACHABLE\$	This macro reflects the total number of hosts that are currently in an UNREACHABLE state.
\$TOTALHOSTSDOWNUNHANDLED\$	This macro reflects the total number of hosts that are currently in a DOWN state that are not currently being "handled". Unhandled host problems are those that are not acknowledged, are not currently in scheduled downtime, and for which checks are currently enabled.
\$TOTALHOSTSUNREACHABLEUNHANDLED\$	This macro reflects the total number of hosts that are currently in an UNREACHABLE state that are not currently being "handled". Unhandled host problems are those that are not acknowledged, are not currently in scheduled downtime, and for which checks are currently enabled.
\$TOTALHOSTPROBLEMS\$	This macro reflects the total number of hosts that are currently either in a DOWN or an UNREACHABLE state.
\$TOTALHOSTPROBLEMSUNHANDLED\$	This macro reflects the total number of hosts that are currently either in a DOWN or an UNREACHABLE state that are not currently being "handled". Unhandled host problems are those that are not acknowledged, are not currently in scheduled downtime, and for which checks are currently enabled.
\$TOTALSERVICESOK\$	This macro reflects the total number of services that are currently in an OK state.
\$TOTALSERVICESWARNINGS\$	This macro reflects the total number of services that are currently in a WARNING state.
\$TOTALSERVICESCRITICAL\$	This macro reflects the total number of services that are currently in a CRITICAL state.
\$TOTALSERVICESUNKNOWN\$	This macro reflects the total number of services that are currently in an UNKNOWN state.
\$TOTALSERVICESWARNINGUNHANDLED\$	This macro reflects the total number of services that are

	currently in a WARNING state that are not currently being "handled". Unhandled services problems are those that are not acknowledged, are not currently in scheduled downtime, and for which checks are currently enabled.
\$TOTALSERVICESCRITICALUNHANDLED\$	This macro reflects the total number of services that are currently in a CRITICAL state that are not currently being "handled". Unhandled services problems are those that are not acknowledged, are not currently in scheduled downtime, and for which checks are currently enabled.
\$TOTALSERVICESUNKNOWNUNHANDLED\$	This macro reflects the total number of services that are currently in an UNKNOWN state that are not currently being "handled". Unhandled services problems are those that are not acknowledged, are not currently in scheduled downtime, and for which checks are currently enabled.
\$TOTALSERVICEPROBLEMS\$	This macro reflects the total number of services that are currently either in a WARNING, CRITICAL, or UNKNOWN state.
\$TOTALSERVICEPROBLEMSUNHANDLED\$	This macro reflects the total number of services that are currently either in a WARNING, CRITICAL, or UNKNOWN state that are not currently being "handled". Unhandled services problems are those that are not acknowledged, are not currently in scheduled downtime, and for which checks are currently enabled.

## Notification Macros

\$NOTIFICATIONTYPE\$	A string identifying the type of notification that is being sent ("PROBLEM", "RECOVERY", "ACKNOWLEDGEMENT", "FLAPPINGSTART", "FLAPPINGSTOP", "FLAPPINGDISABLED", "DOWNTIMESTART", "DOWNTIMEEND", or "DOWNTIMECANCELLED").
\$NOTIFICATIONRECIPIENT\$	A comma-separated list of the short names of all contacts that are being notified about the host or service.
\$NOTIFICATIONISESCALATED\$	An integer indicating whether this was sent to normal contacts for the host or service or if it was escalated. 0 = Normal (non-escalated) notification , 1 = Escalated notification.
\$NOTIFICATIONAUTHOR\$	A string containing the name of the user who authored the notification. If the \$NOTIFICATIONTYPE\$ macro is set to "DOWNTIMESTART" or "DOWNTIMEEND", this will be the name of the user who scheduled downtime for the host or service. If the \$NOTIFICATIONTYPE\$ macro is "ACKNOWLEDGEMENT", this will be the name of the user who acknowledged the host or service problem. If the \$NOTIFICATIONTYPE\$ macro is "CUSTOM", this will be name of the user who initiated the custom host or service notification.
\$NOTIFICATIONAUTHORNAME\$	A string containing the short name of the contact (if applicable) specified in the \$NOTIFICATIONAUTHOR\$ macro.
\$NOTIFICATIONAUTHORALIAS\$	A string containing the alias of the contact (if applicable) specified in the \$NOTIFICATIONAUTHOR\$ macro.

\$NOTIFICATIONCOMMENT\$	A string containing the comment that was entered by the notification author. If the \$NOTIFICATIONTYPE\$ macro is set to "DOWNTIMESTART" or "DOWNTIMEEND", this will be the comment entered by the user who scheduled downtime for the host or service. If the \$NOTIFICATIONTYPE\$ macro is "ACKNOWLEDGEMENT", this will be the comment entered by the user who acknowledged the host or service problem. If the \$NOTIFICATIONTYPE\$ macro is "CUSTOM", this will be comment entered by the user who initiated the custom host or service notification.
\$HOSTNOTIFICATIONNUMBER\$	The current notification number for the host. The notification number increases by one (1) each time a new notification is sent out for the host (except for acknowledgements). The notification number is reset to 0 when the host recovers ( <i>after</i> the recovery notification has gone out). Acknowledgements do not cause the notification number to increase, nor do notifications dealing with flap detection or scheduled downtime.
\$HOSTNOTIFICATIONID\$	A unique number identifying a host notification. Notification ID numbers are unique across both hosts and service notifications, so you could potentially use this unique number as a primary key in a notification database. Notification ID numbers should remain unique across restarts of the Nagios process, so long as you have state retention enabled. The notification ID number is incremented by one (1) each time a new host notification is sent out, and regardless of how many contacts are notified.
\$SERVICENOTIFICATIONNUMBER\$	The current notification number for the service. The notification number increases by one (1) each time a new notification is sent out for the service (except for acknowledgements). The notification number is reset to 0 when the service recovers ( <i>after</i> the recovery notification has gone out). Acknowledgements do not cause the notification number to increase, nor do notifications dealing with flap detection or scheduled downtime.
\$SERVICENOTIFICATIONID\$	A unique number identifying a service notification. Notification ID numbers are unique across both hosts and service notifications, so you could potentially use this unique number as a primary key in a notification database. Notification ID numbers should remain unique across restarts of the Nagios process, so long as you have state retention enabled. The notification ID number is incremented by one (1) each time a new service notification is sent out, and regardless of how many contacts are notified.

## Date/Time Macros

\$LONGDATETIMES\$	Current date/time stamp (i.e. <i>Fri Oct 13 00:30:28 CDT 2000</i> ). Format of date is determined by <a href="#">date_format</a> directive.
\$SHORTDATETIMES\$	Current date/time stamp (i.e. <i>10-13-2000 00:30:28</i> ). Format of date is determined by <a href="#">date_format</a> directive.
\$DATES\$	Date stamp (i.e. <i>10-13-2000</i> ). Format of date is determined by <a href="#">date_format</a> directive.
\$TIMES\$	Current time stamp (i.e. <i>00:30:28</i> ).
\$TIMETS\$	Current time stamp in <code>time_t</code> format (seconds since the UNIX epoch).
\$ISVALIDTIME:\$	This is a special on-demand macro that returns a 1 or 0 depending on whether or not a particular time is valid within a specified timeperiod. There are two ways of using this macro:

	<ol style="list-style-type: none"> <li>1. <b>\$ISVALIDTIME:24x7\$</b> will be set to "1" if the current time is valid within the "24x7" timeperiod. If not, it will be set to "0".</li> <li>2. <b>\$ISVALIDTIME:24x7:timestamp\$</b> will be set to "1" if the time specified by the "timestamp" argument (which must be in time_t format) is valid within the "24x7" timeperiod. If not, it will be set to "0".</li> </ol>
\$NEXTVALIDTIME:\$	<p>This is a special on-demand macro that returns the next valid time (in time_t format) for a specified timeperiod. There are two ways of using this macro:</p> <ol style="list-style-type: none"> <li>1. <b>\$NEXTVALIDTIME:24x7\$</b> will return the next valid time - from and including the current time - in the "24x7" timeperiod.</li> <li>2. <b>\$NEXTVALIDTIME:24x7:timestamp\$</b> will return the next valid time - from and including the time specified by the "timestamp" argument (which must be specified in time_t format) - in the "24x7" timeperiod.</li> </ol> <p>If a next valid time cannot be found in the specified timeperiod, the macro will be set to "0".</p>

## File Macros

\$MAINCONFIGFILES	The location of the <a href="#">main config file</a> .
\$STATUSDATAFILES	The location of the <a href="#">status data file</a> .
\$COMMENTDATAFILES	The location of the comment data file.
\$DOWNTIMEDATAFILES	The location of the downtime data file.
\$RETENTIONDATAFILES	The location of the <a href="#">retention data file</a> .
\$OBJECTCACHEFILES	The location of the <a href="#">object cache file</a> .
\$TEMPFILES	The location of the <a href="#">temp file</a> .
\$TEMPPATHS	The directory specified by the <a href="#">temp path</a> variable.
\$LOGFILES	The location of the <a href="#">log file</a> .
\$RESOURCEFILES	The location of the <a href="#">resource file</a> .
\$COMMANDFILES	The location of the <a href="#">command file</a> .
\$HOSTPERFDATAFILES	The location of the host performance data file (if defined).
\$SERVICEPERFDATAFILES	The location of the service performance data file (if defined).

## Misc Macros

\$PROCESSSTARTTIMES	Time stamp in time_t format (seconds since the UNIX epoch) indicating when the Nagios process was last (re)started. You can determine the number of seconds that Nagios has been running (since it was last restarted) by subtracting \$PROCESSSTARTTIMES\$ from <a href="#">\$TIMETS\$</a> .
---------------------	---

\$EVENTSTARTTIME\$	Time stamp in time_t format (seconds since the UNIX epoch) indicating when the Nagios process starting process events (checks, etc.). You can determine the number of seconds that it took for Nagios to startup by subtracting \$PROCESSSTARTTIME\$ from \$EVENTSTARTTIME\$.
\$ADMINEMAIL\$	Global administrative email address. This value is taken from the <a href="#">admin_email</a> directive.
\$ADMINPAGER\$	Global administrative pager number/address. This value is taken from the <a href="#">admin_pager</a> directive.
\$ARGn\$	The <i>n</i> th argument passed to the command (notification, event handler, service check, etc.). Nagios supports up to 32 argument macros (\$ARG1\$ through \$ARG32\$).
\$USERn\$	The <i>n</i> th user-definable macro. User macros can be defined in one or more <a href="#">resource files</a> . Nagios supports up to 32 user macros (\$USER1\$ through \$USER32\$).