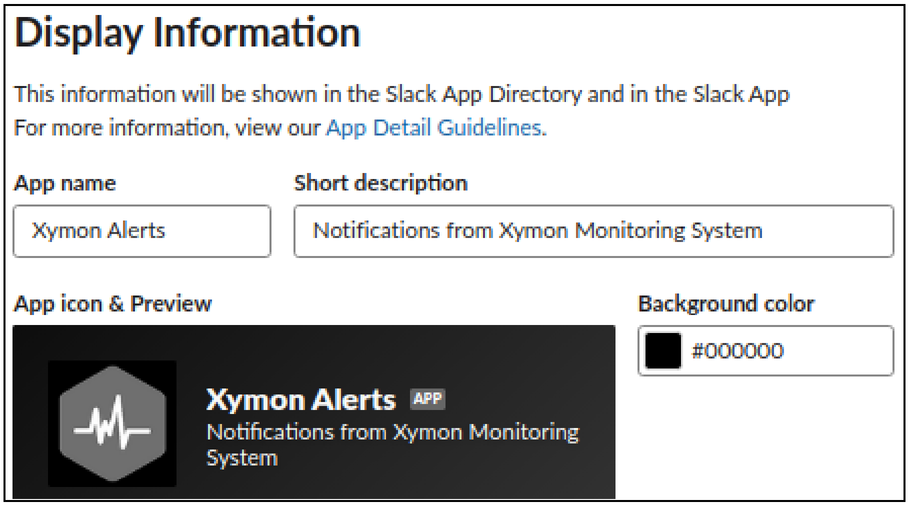
To customize Slack alerts

https://github.com/techspence/xymontoslack
https://api.slack.com/reference/surfaces/formatting

On Slack API website

You must first create a Slack App.

1. Log into your Slack workspace at https://slack.com/workspace-signin
2. Go to https://api.slack.com
3. Click ‘**Your Apps**’ in the top right corner. If this is your first time you’ll probably need to click ‘**Build**’
4. Click ‘**Create an App**’ button
   - Choose ‘from scratch’
   - App Name: Xymon Alerts
   - Workspace: pick yours
5. Click ‘**OAuth & Permissions**’ from the sidebar
6. Scroll down to the ‘**Bot Token Scopes**’ section and click ‘**Add an OAuth Scope**’ button and choose
   - chat:write
   - chat:write:customize
   - users:write
7. Optional: If you wish to restrict the use of this app so it only works from your Xymon Server’s IP range, scroll down to the ‘**Restrict API Token Usage**’ section and define your Xymon Server’s WAN IP, then click the ‘**Save IP address ranges**’ button.
8. Scroll up and click the ‘**Install to Workspace**’ button. You’ll be asked to ‘**Allow**’ the app in your workspace.
9. You now have the ‘**Bot User OAuth Token**’ that’s required in the alert script below.
10. You can also customize the app’s avatar icon and description if you wish. That’s in the ‘**Basic Information**’ section found on the left sidebar menu. I suggest this.



Display Information

This information will be shown in the Slack App Directory and in the Slack App
For more information, view our App Detail Guidelines.

App name: Xymon Alerts

Short description: Notifications from Xymon Monitoring System

App icon & Preview

Background color: #000000

In Slack interface

1. Create new channel where alerts will be sent. It can be private or public. We’ll call ours **xymon-alerts**
2. Add the new app to the channel by typing **/invite** and choosing ‘**Add apps to this channel**’ and searching for your app name

---

On Xymon Server

Note: requires curl package. If it doesn't work, try installing curl

```
# sudo apt install curl
```

1. Create the script. You will need to paste in the '**Bot User OAuth Token**' you got earlier.

   ```
   $ sudo bash
   $ cd /home/xymon/server/bin      ( or wherever your bin folder lives, possibly /usr/lib/xymon/server/bin )
   $ nano slack_app_bot_script.sh
   ```

   ```
   #!/bin/sh
   #--------------------------------
   # Original script from https://github.com/techspence/xymontoslack
   # Rewritten and made awesome by Kris Springer
   # https://www.krisspringer.com
   # https://www.ionetworkadmin.com
   #
   # This script takes 1 parameter (channel name, $channel) and sends Xymon alerts to the specified Slack channel.
   # Assumptions: Slack App with the Bots feature & functionality enabled and a Bot User OAuth Access Token

   hostname=$BBHOSTNAME
   level=$BBCOLORLEVEL
   svcname=$BBSVCNAME
   username="Xymon Alert"
   channel="#${RCPT}"

   NOTICE=`echo "$BBALPHAMSG" | grep -c INFO`
   if [ $RECOVERED -eq 1 ]; then
     alertemoji=":thumbsup:";
     verbiage="Recovered"
   elif [ $NOTICE -gt 0 ]; then
     alertemoji=":waving_white_flag:";
     verbiage="Notice"

   elif [ $level = "red" ]; then
     alertemoji=":rotating_light:";
     verbiage="Down"
   elif [ $level = "yellow" ]; then
     alertemoji=":warning:";
     verbiage="Warning"
   elif [ $level = "green" ]; then
     alertemoji=":thumbsup:";
     verbiage="OK"
   elif [ $level == "purple" ]; then
     alertemoji=":question:";
     verbiage="Missing"
   else
     alertemoji=":waving_white_flag:";
     verbiage="Notice"
   fi

   postdata()
   {
     cat <<EOF
   {
     "channel": "$channel",
     "username": "$username",
     "text": "$alertemoji <$XYMONWEBHOST$XYMONSERVERCGIURL/svcstatus.sh?HOST=$BBHOSTNAME&SERVICE=$BBSVCNAME|*$svcname* $verbiage on $hostname>"
   }
   EOF
   }

   curl --silent --output /dev/null -X POST -H 'Authorization: Bearer paste-token-here' \
     -H 'Content-type: application/json' \
     --data "$(postdata)" "https://slack.com/api/chat.postMessage"
   ```

```
$ chmod 755 slack_app_bot_script.sh
~~$ chown xymon:xymon slack_app_bot_script.sh~~
```

2. In **alerts.cfg** make the following change. Remove MAIL and add SCRIPT and the path. This is where you define the Slack Channel name Instead of an email address.

   ```
   SCRIPT /home/xymon/server/bin/slack_app_bot_script.sh xymon-alerts COLOR=red,yellow,purple DURATION>15m
   RECOVERED NOTICE REPEAT=2h
   ```

3. Trigger a test alert. You can disable some test for a few minutes, or add a non-existing process to the analysis.cfg that will trip a red alert. If it works, you're done!

Here’s a Test script that you can manually run to test things out if you wish to try some additional customizations. You’'ll need to **chmod** and **chown** it just like the production script. You’ll also need the ‘**Bot User OAuth Token**’

```
------------------------
#!/bin/sh
# To run
# sudo ./slack_app_bot_script-TEST.sh

level="yellow"
hostname="TestBox"
svcname="attitude"
username="Test Alert"
channel="#xymon-alerts"

if [ $level = "red" ]; then
  alertemoji=":rotating_light:"
elif [ $level = "yellow" ]; then
  alertemoji=":warning:"
elif [ $level = "green" ]; then
  alertemoji=":waving_white_flag:"
elif [ $level = "purple" ]; then
  alertemoji=":question:"
else
  alertemoji=":waving_white_flag:"
fi

postdata()
{
  cat <<EOF
{
  "channel": "$channel",
  "username": "$username",
  "text": "$alertemoji <https://xymon.innovateteam.com|*$svcname* on $hostname>"
}
EOF
}

# Enable verbose output for debugging
curl --show-error --fail -X POST -H 'Authorization: Bearer paste-token-here' \
  -H 'Content-type: application/json' \
  --data "$(postdata)" "https://slack.com/api/chat.postMessage"

------------------------
```