

# The XY Problem

See Also: [Asking Smart Questions](#)

## What is it?

The XY problem is asking about your attempted *solution* rather than your actual *problem*. This leads to enormous amounts of wasted time and energy, both on the part of people asking for help, and on the part of those providing help.

- User wants to do X.
- User doesn't know how to do X, but thinks they can fumble their way to a solution if they can just manage to do Y.
- User doesn't know how to do Y either.
- User asks for help with Y.
- Others try to help user with Y, but are confused because Y seems like a strange problem to want to solve.
- After much interaction and wasted time, it finally becomes clear that the user really wants help with X, and that Y wasn't even a suitable solution for X.

The problem occurs when people get stuck on what they believe is the solution and are unable step back and explain the issue in full.

## What to do about it?

1. Always include information about a broader picture along with any attempted solution.
2. If someone asks for more information, do provide details.
3. If there are other solutions you've already ruled out, share why you've ruled them out. This gives more information about your requirements.

Remember that if your diagnostic theories were accurate, you wouldn't be asking for help right?

## Examples

### Example 1

n00b doesn't actually want the last 3 chracters in a filename, he wants the file extensions, so why ask for the last 3 characters?

```
<n00b> How can I echo the last three characters in a filename?
<feline> If they're in a variable: echo ${foo: -3}
<feline> Why 3 characters? What do you REALLY want?
<feline> Do you want the extension?
<n00b> Yes.
<feline> There's no guarantee that every filename will have a three-
letter extension,
<feline> so blindly grabbing three characters does not solve the
```

```
problem.  
<feline> echo ${foo##*.}
```

## Example 2

If Angela had just started by explaining she wants to prevent others from detecting her OS, this could have been a much shorter and more productive discussion.

Angela: 'nmap -O -A 127.0.0.1' returns some lines starting with 'OS:'. How to change it?  
Obama: Look in the sourcecode for nmap, find how it figures out the Linux part, then rewrite your TCP/IP stack to not operate in a way nmap can detect.  
Angela: Yeah, but I don't know about linux system api at all.  
Obama: Well, nmap's fingerprint is based on the way the TCP/IP stack works, there's no real way except to rewrite the appropriate parts of said stack.  
Angela: I really need to avoid these messages. Can iptables do this work?  
Obama: Well, don't use OS detection or version scanning  
Angela: I want to prevent others from knowing the type of my OS

[Source 1](#) ... [Source 2](#)