**The difference between X-Forwarded-For and X-Real-IP?**

Recommended for you: [Get network issues from **WhatsUp Gold**. **Not end users.**](http://shareasale.com/r.cfm?b=1080234&u=1121089&m=75001&urllink=&afftrack=)

**X-Forwarded-For and X-Real-IP areas and obtain a client IP?**  
  
Generally speaking, X-Forwarded-For is used to record the information of each agent, after the first level agent (anonymous proxy exception), the proxy server will send the request additional sources of IP in X-Forwarded-For  
  
A request from 4.4.4.4, header contains a line  
  
X-Forwarded-For: 1.1.1.1, 2.2.2.2, 3.3.3.3  
On behalf of the request issued by 1.1.1.1, after three layers of agent, the first layer is 2.2.2.2, the second layer is 3.3.3.3, and the source of IP4.4.4.4 the request is third layer proxy  
  
X-Real-IP, generally only truthful record of the requesting client IP, the above example, if the configuration of the X-Read-IP, will be  
  
X-Real-IP: 1.1.1.1  
So, if there is only one layer of agency, value is the same as the two heads  
  
The first

1. /\*\*
2. \* To obtain client IP from a Request object, processing the reverse proxy to intercept HTTP proxy server and Nginx IP
3. \* @param request
4. \* @return ip
5. \*/
6. **public** **static** String getLocalIp(HttpServletRequest request) {
7. String remoteAddr = request.getRemoteAddr();
8. String forwarded = request.getHeader("X-Forwarded-For");
9. String realIp = request.getHeader("X-Real-IP");
11. String ip = **null**;
12. **if** (realIp == **null**) {
13. **if** (forwarded == **null**) {
14. ip = remoteAddr;
15. } **else** {
16. ip = remoteAddr + "/" + forwarded.split(",")[0];
17. }
18. } **else** {
19. **if** (realIp.equals(forwarded)) {
20. ip = realIp;
21. } **else** {
22. **if**(forwarded != **null**){
23. forwarded = forwarded.split(",")[0];
24. }
25. ip = realIp + "/" + forwarded;
26. }
27. }
28. **return** ip;
29. }

Second

1. **public** **static** String getIp(HttpServletRequest request) {
2. String remoteAddr = request.getRemoteAddr();
3. String forwarded = request.getHeader("X-Forwarded-For");
4. String realIp = request.getHeader("X-Real-IP");
6. String ip = **null**;
7. **if** (realIp == **null**) {
8. **if** (forwarded == **null**) {
9. ip = remoteAddr;
10. } **else** {
11. ip = remoteAddr + "/" + forwarded;
12. }
13. } **else** {
14. **if** (realIp.equals(forwarded)) {
15. ip = realIp;
16. } **else** {
17. ip = realIp + "/" + forwarded.replaceAll(", " + realIp, "");
18. }
19. }
20. **return** ip;
21. }

Third

1. **public** **static** String getIp2(HttpServletRequest request) {
2. String ip = request.getHeader("X-Forwarded-For");
3. **if**(StringUtils.isNotEmpty(ip) && !"unKnown".equalsIgnoreCase(ip)){
4. //There will be multiple IP values repeatedly reverse proxy, the first IP is true and IP
5. **int** index = ip.indexOf(",");
6. **if**(index != -1){
7. **return** ip.substring(0,index);
8. }**else**{
9. **return** ip;
10. }
11. }
12. ip = request.getHeader("X-Real-IP");
13. **if**(StringUtils.isNotEmpty(ip) && !"unKnown".equalsIgnoreCase(ip)){
14. **return** ip;
15. }
16. **return** request.getRemoteAddr();
17. }