The function *resolve_addr* is used to resolve a hostname into an IP address. It takes a pointer to a *sockaddr* structure and a *hostname* as parameters.

Working:
- It declares two pointers, *address* and *host*. *address* is a pointer to a *sockaddr_in* structure (which is a specific type of *sockaddr* used for IP addresses), and *host* is a pointer to a *hostent* structure (which is used by the *gethostbyname* function).
- It casts the *addr* parameter to a *sockaddr_in* pointer and assigns it to *address*. This is done because the *addr* parameter is a generic *sockaddr* structure, but we need to work with an IP-specific sockaddr_in structure.
- It initializes the *address* structure to zero using *memset*. This is done to ensure that all fields in the structure start with a clean state.
- It sets the *sin_family* field of *address* to *AF_INET*, which indicates an IP address.
- It tries to convert the *hostname* parameter to an IP address using *inet_addr*. If *hostname* is already an IP address, this will succeed and *inet_addr* will return the IP address. Otherwise, it will return -1.
- If *inet_addr* returned -1, it calls *gethostbyname* to resolve the hostname. If *gethostbyname* succeeds, it copies the IP address into the *sin_addr* field of address and returns 0. If *gethostbyname* fails, it returns -1.
- If *inet_addr* did not return -1, it returns 0. This means that the *hostname* parameter was already an IP address, so no resolution was necessary.

The changes made to *address* is reflected on *addr.*

In summary, resolve_addr is a function that takes a hostname and a pointer to a sockaddr structure, and fills in the structure with the IP address corresponding to the hostname.